IBM @server

IBM

JD Edwards EnterpriseOne 8.9 with IBM @server iSeries and Storage

- Updated installation tips for JD Edwards EnterpriseOne 8.9 including Unicode
- IASP (switched disk) solutions with JD Edwards EnterpriseOne 8.0
- WebSphere tuning tips for JD Edwards EnterpriseOne 8.9

Randy Erickson
Gerrie Fisk
Marie Gann
Brian Porter
Maruf Rafik
Clark Scholten
Neil Willis

Redbooks

ibm.com/redbooks

International Technical Support Organization

JD Edwards EnterpriseOne 8.9 with IBM @server iSeries and Storage

August 2005

SG24-6359-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (August 2005)

This edition applies to Version 5, Release 2, of OS/400 (product number 5722-SS1) and JD Edwards EnterpriseOne 8.9.

© Copyright International Business Machines Corporation 2005. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices vii
Trademarks viii

Preface ix
The team that wrote this redbook. ix
Become a published author xi
Comments welcome. xi

Chapter 1. Introduction to this book 1

Chapter 2. JD Edwards EnterpriseOne installation overview 3
2.1 Preinstallation checklist. 4
2.2 Installing the deployment server 9
2.2.1 Loading files on the deployment server 9
2.2.2 Post deployment server installation tasks. 13
2.3 Installing the Enterprise Server. 14
2.3.1 Loading files on the Enterprise Server 14
2.3.2 Loading central objects on the Enterprise Server. 15
2.4 Installing the deployment server 16
2.4.1 Installation plan. 16
2.4.2 Installation Workbench 16
2.5 Installing the Enterprise Server. 18
2.5.1 Modifying the Enterprise Server jde.ini file 18
2.5.2 Modifying the library file that resides on the deployment server. 18
2.5.3 Installing the server libraries. 19
2.5.4 Creating an output queue for the iSeries default printer. 20
2.5.5 Creating the job queue 20
2.5.6 Running the PORTTEST program 21
2.5.7 Starting EnterpriseOne services. 21
2.5.8 Setting up an EnterpriseOne printer. 22
2.6 Installing client workstations 22
2.7 Completing the install 23

Chapter 3. JD Edwards EnterpriseOne 8.9 performance management 25
3.1 Tuning Apache HTTP server performance 26
3.1.1 Measuring and analyzing performance. 26
3.1.2 Adjusting settings to improve performance. 27
3.2 Tuning WebSphere Application Server performance 35
3.2.1 Measuring and analyzing application and JAS servers. 35
3.2.2 Adjusting settings 37
3.3 Tuning EnterpriseOne application performance 41
3.3.1 Measuring and analyzing performance. 41
3.3.2 Adjusting configuration parameters. 42
3.4 Tuning iSeries database performance 44
3.4.1 Measuring and analyzing performance. 45
3.4.2 Adjusting DB2 UDB parameter settings 47

Chapter 4. Upgrading to JD Edwards EnterpriseOne 8.9 Unicode on iSeries Servers 53
4.1 The test environment 54

© Copyright IBM Corp. 2005. All rights reserved.

4.2 JD Edwards EnterpriseOne ERP 8.0 55
4.3 The upgrade process 55
4.4 Upgrading to JD Edwards EnterpriseOne 8.9 (non-Unicode) 55
4.5 JD Edwards EnterpriseOne 8.9 Unicode conversion 59
4.6 Relative performance 61
4.7 Summary 62

Chapter 5. Installing the deployment server on Integrated xSeries Server for iSeries 63
5.1 Overview of the Integrated xSeries Server installation 64
5.2 Setting up the operating environment 66
5.3 Installing Windows 2000 or Windows 2003 server 66
5.3.1 Adding Network server storage spaces for JD Edwards implementation 69
5.3.2 Operational tips 70

Chapter 6. Virtualization for server and application consolidation 71
6.1 Virtualization benchmark overview 72
6.2 Virtualization benchmark goals 72
6.3 Application and server workload description 74
6.3.1 JD Edwards Enterprise One 8.9 application 74
6.3.2 JD Edwards Enterprise Learning Management 8.81 application 74
6.3.3 Trade3 application 74
6.3.4 Samba file sever 75
6.3.5 Virtualization benchmark methodology 75
6.4 System architecture 76
6.4.1 Virtualization system configuration 76
6.4.2 Network configuration 77
6.4.3 Virtualization Engine Console 78
6.4.4 Virtualization benchmark performance results 79
6.5 Summary 83

Chapter 7. Switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment 85
7.1 Hardware environment architecture 86
7.2 Software environment architecture 87
7.3 Configuring the IASP for JD Edwards EnterpriseOne 88
7.4 Moving JD Edwards EnterpriseOne libraries to the IASP 89
7.4.1 Saving the libraries into save files 89
7.4.2 Deleting the libraries from the system ASP 89
7.4.3 Restoring the database libraries to the IASP 90
7.5 Moving the JD Edwards EnterpriseOne IFS directories to the IASP 90
7.5.1 Coping the directories to the IASP 90
7.5.2 Creating symbolic links 92
7.6 Configuring JD Edwards EnterpriseOne data sources and OCM tables 93
7.7 Configuring JD Edwards EnterpriseOne user profiles 95
7.7.1 Creating a job description 95
7.7.2 Adding the new job description to user profiles 96
7.8 Editing the JD Edwards EnterpriseOne enterprise server's jde.ini file 97
7.9 Adding a host table entry for the takeover IP 98
7.10 Configuring JD Edwards EnterpriseOne fat clients 98
7.10.1 Changing the ODBC connections 99
7.10.2 Network connections 100
7.11 Configuring the JD Edwards EnterpriseOne Java Application Server 101
7.12 Switching between nodes 101
7.12.1 Performance considerations when switching nodes 102

7.13 Further resources 102

Chapter 8. Storage area network technology 103
8.1 Overview of SAN technology 104
8.2 Overview of IBM SAN attached disk storage 105
8.2.1 IBM TotalStorage DS6000 series 105
8.2.2 TotalStorage DS8000 series 105
8.2.3 IBM TotalStorage Enterprise Storage Server 106
8.3 IBM TotalStorage Resiliency Family for iSeries servers 107
8.3.1 IBM TotalStorage FlashCopy 107
8.3.2 IBM Total Storage peer-to-peer-remote-copy 108
8.4 IBM @server i5 and iSeries with IBM SAN Disk 110
8.5 Why iSeries and SAN 111
8.6 Performance effects between the iSeries and SAN 112
8.6.1 Interoperability concerns 112
8.7 iSeries and SAN technology considerations 113
8.7.1 Internal versus external storage 113
8.7.2 iSeries server expert cache 113
8.7.3 SAN storage cache 114
8.7.4 SAN configuration: Dedicated versus shared disk 114
8.7.5 Multi-pathing from iSeries to SAN 115
8.7.6 Connectivity of iSeries to SAN 115
8.8 JD Edwards EnterpriseOne and JD Edwards World with SAN advantages 116
8.9 JD Edwards EnterpriseOne and JD Edwards World implementation considerations . 117
8.10 IBM TotalStorage SAN Disk with JD Edwards 117
8.11 Summary 118

Appendix A. The JD Edwards EnterpriseOne 8.9 test environment and transaction details 119
JD Edwards EnterpriseOne 8.9 Web server test environment 120
Virtualization benchmark JD Edwards Enterprise One 8.9 transaction details 121
JD Edwards Enterprise Learning Management 8.81 transaction details 123

Appendix B. JD Edwards EnterpriseOne 8.9 test configuration files 125
HTTPD.CONF file 126
The jas.ini file 127
The JDBj.INI file 130
The jde.ini file 137

Appendix C. Additional information for switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment 145
Relational database directory entry 146
Synchronizing user profiles across nodes 148

Related publications 153
IBM Redbooks 153
Other publications 153
Online resources 154
How to get IBM Redbooks 155
Help from IBM 155

Index 157

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

© Copyright IBM Corp. 2005. All rights reserved.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

@server®
@server®
Redbooks (logo) ™
iSeries™
i5/OS™
pSeries®
xSeries®
zSeries®
Advanced Micro-Partitioning™
AIX 5L™
AIX®
AS/400®
Domino®
DB2 Universal Database™
DB2®
Enterprise Storage Server®
ESCON®
FlashCopy®
IBM®
Lotus®
Micro-Partitioning™
NetServer™
Operating System/400®
OS/400®
PartnerWorld®
PowerPC®
POWER™
POWER4™
POWER5™
Redbooks™
Tivoli®
TotalStorage®
Virtualization Engine™
WebSphere®

The following terms are trademarks of other companies:

IPC, Java, JavaServer, JavaServer Pages, JDBC, JVM, J2EE, Sun, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft Internet Explorer, Microsoft, Visual C++, Windows server, Windows NT, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Itanium, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Preface

JD Edwards EnterpriseOne is a complex application that provides Enterprise Resource Planning (ERP), Supply Chain Management (SCM), and Customer Relationship Management (CRM) capabilities. It can run with Citrix, fat, or Web clients.

This IBM® Redbook provides helpful hints as you work with the EnterpriseOne product on iSeries™ in general. It specifically includes topics for EnterpriseOne 8 and 8.9. It brings together several varied topics into one reference guide.

The reader of this book should have a basic knowledge of IBM @server® iSeries and the JD Edwards EnterpriseOne solution.

This book is designed to assist technical people among J.D. Edwards OneWorld customers, OneWorld consultants, business partners, and IBM service representatives. It targets professionals who are directly involved with the implementation of a total business solution, consisting of iSeries server hardware, OS/400®, DB2® UDB for iSeries database, OneWorld solutions, and supplemental solution products.

You should use this book in conjunction with *J.D. Edwards OneWorld Implementation for AS/400*, SG24-5195. This publication explains detailed concepts and all implementation steps of earlier versions of OneWorld, most of which still apply to OneWorld Xe. You should also use this book with the J.D. Edwards manuals that are provided with the OneWorld software.

The IBM team and J.D. Edwards team have worked closely together since the introduction of the AS/400® in 1988 to bring customers a superior business solution. That partnership continued through PeopleSoft's acquisition of J.D. Edwards and, more recently, continues through Oracle's acquisition of PeopleSoft in early 2005. IBM is solidly behind the EnterpriseOne product line and the joint customers that this solution represents.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization (ITSO), Rochester Center.

Randy Erickson, is a Staff Software Engineer with IBM in Rochester. Randy joined IBM in 1998, working on the DB2 Database Host Server team for the IBM @server iSeries system. Since 2003, he has worked in development, supporting JD Edwards EnterpriseOne and IBM i5/OS™ development organizations. Prior to IBM, Randy worked as a application development architect in the telecommunications industry. Randy holds a Bachelor of Science degree in Computer Science from the University of Wisconsin.

Gerrie Fisk is a Senior I/T Specialist in Denver, Colorado, and has been with IBM for 27 years. She is a member of the IBM and Oracle International Competency Center, specializing in the iSeries server. Gerrie is responsible for EnterpriseOne benchmarks and sizing tests on the iSeries server. Her previous assignments were in the midrange arena as an IBM Systems Engineer installing, supporting, and upgrading over 500 AS/400s and S/3Xs, as well as teaching many AS/400 classes and seminars.

© Copyright IBM Corp. 2005. All rights reserved.

Marie Gann is a software engineer with IBM in Rochester and is part of the ERP development team. Her primary focus is Oracle and the EnterpriseOne product. She is also part of the development team for the iSeries file system functions. Marie has most recently worked with Oracle on their support of independent ASPs.

Brian Porter is an I/T Specialist for the IBM Advanced Technical Support Storage Solutions Group. He has 12 years of experience in the UNIX® systems, storage, and services industry. He joined IBM five years ago and spent the first four years in IBM Global Services, focused on client AIX® systems and storage solutions on IBM @server pSeries®. He currently supports the ATS Solutions-Storage Alliance ISVs. He holds a Bachelor of Science degree in Electrical and Computer Engineering with a minor in Mathematics from Brigham Young University, Utah.

Maruf Rafik is a Senior Technical Consultant and has been with Oracle for over four years. He is currently focused on providing technical consulting and support for EnterpriseOne software on the iSeries platform. His responsibilities include working extensively with the IBM WebSphere® product family running on multiplatform and the EnterpriseOne Web product. Maruf is a graduate in Computer Information Science and holds industry certifications from IBM on DB2 UDB V7.1 and WebSphere Application Server V4.0.

Clark Scholten is a Senior Software Engineer with 17 years of experience with IBM. He has worked as a development liaison between JD Edwards EnterpriseOne and IBM i5/OS development organizations for the last five years. Previously, Clark has worked in iSeries Access development for nine years and in PartnerWorld® doing Java™ education for three years.

Neil Willis is an Advisory Software Engineer with IBM in Rochester. Neil has worked in IBM Rochester for nine years in the areas of application development, performance design, and benchmarking. Previously, he was an Advisory Systems Engineer with IBM Australia for 10 years. His current projects include high-level scaling of Web-based commercial software applications and unicode performance. He has a Masters Degree in Information Systems.

Thanks to the following people for their contributions to this project:

Matthew Bedernjak
TotalStorage® IT Specialist
IBM Toronto

Leslie Clark
IBM @server Solutions Relationship Manager for JD Edwards EnterpriseOne and World Software
IBM Rochester

Boyd Fenton
Oracle International Competency Center Manager
IBM Denver

Susan Powers
ITSO Project Manager, Rochester Center
IBM Rochester

Julie Ransom
iSeries Alliance Manager to Oracle
IBM Rochester

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners, or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. jlu Building 107-2
  3605 Highway 52N
  Rochester, Minnesota 55901-7829

1

Introduction to this book

To use this book effectively, you should have a basic knowledge of IBM @server iSeries and the JD Edwards EnterpriseOne solution. We designed this book to assist technical people among J.D. Edwards OneWorld customers, OneWorld consultants, business partners, and IBM service representatives. It targets professionals who are directly involved with the implementation of a total business solution, consisting of iSeries server hardware, OS/400, DB2 UDB for iSeries database, OneWorld solutions, and supplemental solution products.

Note: You should use this book in conjunction with *J.D. Edwards OneWorld Implementation for AS/400*, SG24-5195. This publication explains detailed concepts and all implementation steps of earlier versions of OneWorld, most of which still apply to OneWorld Xe. You should also use this book with the J.D. Edwards manuals that are provided with the OneWorld software.

In addition to this introductory chapter, this book includes the following chapters:

- Chapter 2, “JD Edwards EnterpriseOne installation overview” on page 3

  Provides an overview of the basic installation of JD Edwards EnterpriseOne when using:

  - An iSeries server with OS/400 V5R2 as an enterprise server
  - An Intel®-Pentium based Server with Windows® as a deployment server

- Chapter 3, “JD Edwards EnterpriseOne 8.9 performance management” on page 25

  Provides an overview of performance management, starting with the fundamentals of system tuning. The chapter goes on to describe WebSphere application server tuning, EnterpriseOne application tuning, and iSeries database tuning.

- Chapter 4, “Upgrading to JD Edwards EnterpriseOne 8.9 Unicode on iSeries Servers” on page 53

  Documents the technical experience of upgrading to JD Edwards EnterpriseOne 8.9 non-Unicode on an IBM @server iSeries server, followed by upgrading the database from non-Unicode to Unicode. The results include timing the process of the two upgrades, as well as tracking the changes in the size of the database. Also includes helpful tips and resource planning guidelines.

© Copyright IBM Corp. 2005. All rights reserved.

- Chapter 5, “Installing the deployment server on Integrated xSeries Server for iSeries” on page 63

  Describes what is unique for the installation of the Windows operating system when the deployment server is an integrated part of the iSeries server. Topics include an overview of the Integrated xSeries® Server installation, the setup of the operating environment, and the installation of the Windows server™.

- Chapter 6, “Virtualization for server and application consolidation” on page 71

  Describes how IBM uses new features of the IBM @server i5 server to maximize the performance of four applications running under different operating systems. Also describes the IBM Grand Slam Virtualization benchmark that demonstrates how JD Edwards Enterprise applications can be run on the same server as EnterpriseOne

- Chapter 7, “Switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment” on page 85

  Demonstrates how to set up and operate a switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment. High availability solutions come in various shapes and forms. One of the most straightforward to implement and to use is generically called a *switched disk* or *switchable device topology*.

- Chapter 8, “Storage area network technology” on page 103

  Discusses Storage Area Network (SAN) technology and how it relates to the iSeries servers that have implemented JD Edwards EnterpriseOne or JD Edwards World software solutions. The SAN benefits of simplified storage management, disk virtualization, backup, disaster recovery, and performance, to name a few, are available on the iSeries server, often driven by the storage consolidation of OS/400, Windows, and UNIX platforms. IBM, in particular the iSeries servers, have extended system support of SAN technology. Specifically, this chapter provides Oracle users with a better understanding as to what the SAN technologies are and how they are applicable to Oracle environments.

2

JD Edwards EnterpriseOne installation overview

This chapter provides an overview of the approach for the basic installation of JD Edwards EnterpriseOne when using:

- An iSeries server with OS/400 V5R2 as an enterprise server
- An Intel-Pentium based Server with Windows 2000 as a deployment server

Note: The information that this chapter provides presents a high-level view of the installation process of JD Edwards EnterpriseOne for experienced product installers. As appropriate, this chapter provides supplemental information that is not currently available in the *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* . However, you should use the *Installation PeopleBook* in conjunction with the information that is provided in this chapter. You can access this book with an authorized login through the Oracle PeopleSoft Customer Connection at:

`http://www.peoplesoft.com/corp/en/public_index.jsp`

To navigate to the book, select **Support** → **Documentation** → **Documentation updates**.

© Copyright IBM Corp. 2005. All rights reserved.

2.1 Preinstallation checklist

Prior to the installation, address the steps outlined in this section to help ensure the success of the installation process:

1. The deployment server, enterprise server, and workstations must adhere to the hardware and software requirements mentioned in the Minimum Technical Requirements document. To view this document, go to the Oracle Web site at:

   http://www.peoplesoft.com

   and log on using the *Customer and Partners* connection.

2. Check the disk requirements using the *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* document.

3. Update the iSeries server that is to be used as an enterprise server with the most recent PTFs that are recommended in the Informational APAR found at:

   http://www-1.ibm.com/servers/enable/site/events/jde_links.html

4. Update the deployment server with the iSeries Access service pack suggested in the Informational APAR at:

   http://www-1.ibm.com/servers/enable/site/events/jde_links.html

   It is important that the recommended iSeries Access service pack is installed on the deployment server.

5. Use the Display System Value (**DSPSYSVAL**) command to verify that the iSeries system value parameter settings are set as identified in Table 2-1.

   Table 2-1 iSeries system value settings

System Value	Parameter setting
QRESTORE	*ALL
QALWOBJRST	*ALL
QJOBMSGQMX	64
QJOBMSGQFL	*WRAP or *PRTWRAP

   If the system values are not set up as identified in Table 2-1, use an iSeries user profile with appropriate authority (for example, QSECOFR authority or the QSECOFR user profile). Use the Change System Value command (**CHGSYSVAL**) to change the parameter setting as recommended.

6. Add an entry to set the “LOB_LOCATOR_THRESHOLD” to 10000 in the QAQQINI file which resides in the QUSRSYS LIBRARY on the iSeries server. Without this entry in the QAQQINI file, any process (for example, a Server package build) that fetches more than 250,000 LOBs fails. If the QAQQINI file in QUSRSYS does not reflect it, make the following changes:

   a. Use the Work with Object (**WRKOBJ**) command to open the QAQQINI file:

      ```
      WRKOBJ QUSRSYS/QAQQINI *FILE
      ```

   b. If the QAQQINI file does not exist, use the Create Duplicate Object command (**CRTDUPOBJ**) and add it from the template residing in the QSYS library:

      ```
      CRTDUPOBJ QAQQINI QSYS *FILE QUSRSYS DATA(*YES)
      ```

   c. Enter **STRSQL** from the command prompt to invoke SQL then run the following command:

      ```
      INSERT INTO QUSRSYS/QAQQINI VALUES('LOB_LOCATOR_THRESHOLD','10000', default)
      ```

d. Stop and restart the QUSRWRK subsystem as the new database connections must be started with the new values for the QZDASOINIT value.

For further details, check the PTF Cover Letter of SI08140.

Note: The same changes can also be made using iSeries Access Navigator.

7. Use the Display Software Resource command (**DSPSFWRSC**) to verify that the QSYSINC library is installed. It is installed with the operating system if option 13, OS/400 - System Openness Includes is installed.
8. Make sure the iSeries server has a valid C compiler license. To do so, type the Create Command (CRTCMD) using TEST as a module name

   ```
   CRTCMD MODULE (TEST)
   ```

   If the iSeries compiler license is current, then an error message appears (such as, `File QCSRC with member TEST could not be opened or member TEST could not be found in file QCSRC`). If the license is not current, an error is returned. The compiler license is required to successfully build Business Functions on the iSeries.
9. TCP/IP should already be configured on the iSeries server as part of the infrastructure preparation. Using the QSECOFR user ID, use the Configure TCP command (**CFGTCP**) and select option 10 (Work with TCP/IP host table entries) and option 12 (Change TCP/IP domain information) to verify TCP/IP has been setup correctly to reflect the sample values as listed in Table 2-2:

Table 2-2 TCP/IP settings

CFGTCP - option 10	
Internet Address	10.219.1.19
Host name	DENPS1 DENPS1.PEOPLESOFT.COM
CFGTCP - option 12	
Host name	DENPS1
Domain name	PEOPLESOFT.COM
Host name search priority	*LOCAL

The **CFGTCP** option 12 entry for Host Name for the enterprise server is the same name as the current system name. Use the Display Network Attributes command (**DSPNETA**) to verify it, as shown in Figure 2-1 on page 6.

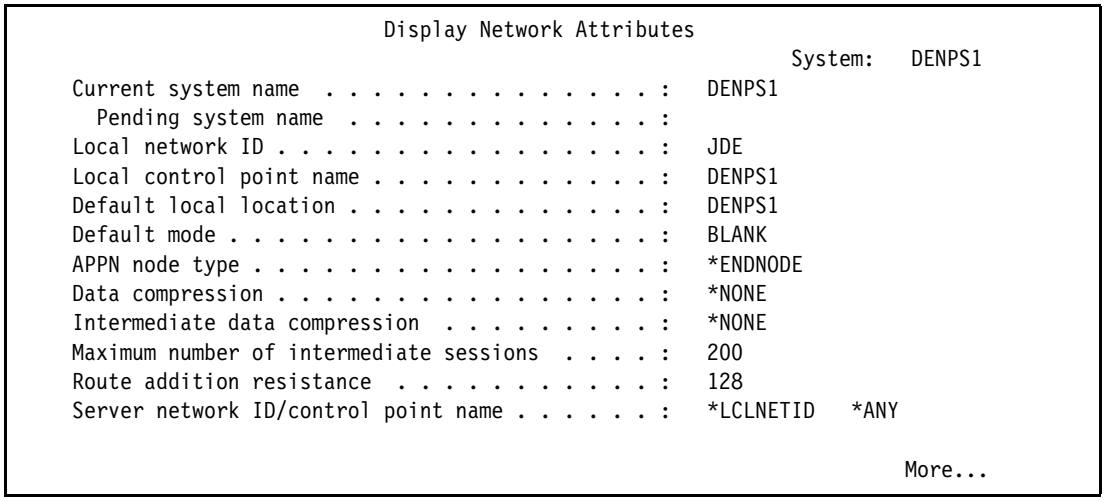

Figure 2-1 DSPNETA

10. Use the Verify TCP/IP Connection command (**PING**) from the iSeries server using the system name, not the TCP/IP address, of the Deployment server. This ensures that successful communications can be established between the two systems. If the PING does not work, create a host table entry for the Deployment server on the iSeries enterprise server.
11. Make sure that the Enterprise server has enough QZDASONIT prestart jobs; otherwise your system can be slow to make connections.

    To determine the number of prestart jobs to set up, consider these facts:

    – Each EnterpriseOne user typically uses three to five Open Database Connectivity (ODBC) connections
    – Each ODBC connection uses one QZDASOINIT program

    Therefore, set the initial value for QZDASOINIT to four times the number of expected concurrent connections.

    Note: Because the determination of the correct number to specify involves a degree of calculation, review this setting again when a typical number of users are accessing the system.

    Enter the Display Subsystem command **(DSPSBSD)** to see the current settings, as follows:

    ```
    DSPSBSD QUSRWRK
    ```

    Choose option 10. Then choose option 5 for program QSYS/QZDASOINIT. The display now shows you the prestart job entry details.

    Prompt (F4) the following Change Prestart Job Entry command (**CHGPJE**) to access the form on which you can change the number of prestart jobs:

    ```
    CHGPJE SBSD(QSYS/QUSRWRK) PGM(QSYS/QZDASOINIT)
    ```

    Change the number of prestart jobs as needed, and then press **Enter.** Restart the QUSRWRK subsystem for the changes to take effect.

12. Use the command **CWBPING *<name of iSeries server>*** from the deployment server via a DOS prompt to make sure database host servers are running successfully in the iSeries server. See the log shown in Figure 2-2.

13. At a DOS prompt on the deployment server, execute the **CWBPING** command to make sure database host servers are running successfully in the iSeries server:

```
CWBPING <name of iSeries server>
```

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator>cwbping DENPS1
IBM AS/400 Client Access Express for Windows
Version 5  Release 1  Level 0
(C) Copyright IBM Corporation and Others 1984, 2000.  All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure
 restricted by GSA ADP Schedule Contract with IBM Corp.
Licensed Materials - Property of IBM

To cancel the CWBPING request, press CTRL-C or CTRL-BREAK
I - Verifying connection to system DENGS1...
I - Successfully connected to server application: Central Client
I - Successfully connected to server application: Network File
I - Successfully connected to server application: Network Print
I - Successfully connected to server application: Data Access
I - Successfully connected to server application: Data Queues
I - Successfully connected to server application: Remote Command
I - Successfully connected to server application: Security
I - Successfully connected to server application: Telnet
I - Successfully connected to server application: Management Central
I - Connection verified to system DENGS1

C:\Documents and Settings\Administrator>
```

Figure 2-2 The cwbping command display

14. Issue the Work with Relational Database Directory Entries command (**WRKRDBDIRE *ALL**) from the iSeries command prompt and make sure that the value is set to the name of the enterprise server and has a remote location name of *LOCAL. If the value is not set, enter the following command on an OS/400 command line, and press Enter:

```
WRKRDBDIRE * ALL
```

See Figure 2-3 for an illustration.

```
               Work with Relational Database Directory Entries

Position to  . . . . . .

Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display details   6=Print details

        Relational          Remote
Option  Database            Location                  Text

        DENPS1              *LOCAL
```

Figure 2-3 WRKRDBDIRE display

If it does not exist, use the Add RDB Directory Entry (**ADDRDBDIRE**) command to add a relational database directory entry on the iSeries enterprise server. This entry is used to establish ODBC connections to the EnterpriseOne data tables residing on the iSeries server. If the directory entry is not on this list, enter the following command on an OS/400 command line where system name is the name of your iSeries server.

```
ADDRDBDIRE RDB(system name) RMTLOCNAME(*LOCAL) TYPE(*IP)
```

Press **Enter**.

15. During the installation process, the user logs into the iSeries server as QSECOFR several times. The installation process attempts to change the QSECOFR message queue to *BREAK mode so that the operator is prompted to load the CDs in sequence.

    Note: It is very important that only one user is logged in as QSECOFR at that time. If anyone else is logged on as QSECOFR the message queue can be locked and the EnterpriseOne on iSeries installation process can fail.

16. Most of the EnterpriseOne 8.9 UBEs need two database connections. The second connection is a socket connection and goes through the Extended Dynamic Remote SQL (EDRSQL) server, even on the same system. It is therefore required that the EDRSQL server application is running on the iSeries enterprise server.

    Use the following command to start the EDRSQL application server if it is not running:

    ```
    STRTCPSVR SERVER(*EDRSQL)
    ```

17. Set the Windows 2000 Server paging file size on the deployment server to optimize system performance. It is recommended to set the Initial Size for the Paging File Size field to a number 1.5 times the amount of RAM available on the system. Again, set the Maximum Size to around double the initial size. However, do not set the maximum number too high on systems where there is limited overall disk space, as this has a negative affect on system performance.

18. If you have Microsoft® SQL Server 2000 Client Tools or Enterprise Manager installed on the deployment server, there is a known Microsoft issue that can cause the installation of the deployment server to fail. The failure is due to the Microsoft SQL Server Desktop Engine (MSDE) component using the same .dll as the SQL Server 2000 database.

    To avoid this failure, rename a few .dll and .rll files (specifically, the Semnt.dll, Sqlsvc.dll, Sqlresld.dll, Semnt.rll, and Sqlsvc.rll files). For details on the actions to take, refer to the “Considerations for MSDE Installation” section in *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

19. Make sure that the JDE user ID that is created is part of the Administrators group on the deployment server has a password of JDE. The password is case-sensitive.

20. Make sure FTP services is installed and set up properly so that you can perform FTP from the Deployment server to the enterprise server. The File Transfer Protocol (FTP) is used during the Installation Workbench and host code load processes. For details on how to set it up, refer to the “Configuring the Deployment Server” section in *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

21. Ensure that supported Microsoft Visual C++® version and service pack is installed on the deployment server.

22. Use iSeries Navigator on the deployment server PC to properly define a connection for the iSeries enterprise server. On the “Connection” page of the “Server properties” panel for

the “My Connections” definition, setup the following configuration parameters to improve the performance:

- *IP Address lookup frequency*: After startup
- *Where to lookup remote port*: Standard

23. If the Install site's Information Technology (IT) policy allows disabling the Virus Checker, turn it off on the Deployment server during the install to speed up the process. If the IT policy doses not allow doing so, at least make sure to place \PeopleSoft directory in the exclusion list to avoid being scanned.

2.2 Installing the deployment server

The deployment server acts as the focal point for the EnterpriseOne installation process and the ongoing creation and deployment of EnterpriseOne packages to server and client workstations. Oracle requires the deployment server to be an Intel-based server that is running the Windows operating system.

Note: Oracle supports an IBM Integrated xSeries Server that you can use as the deployment server if you are running the Windows operating system. The guide *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* includes an overview of the Integrated xSeries Server installation.

In this book, you can find additional information in Chapter 5, “Installing the deployment server on Integrated xSeries Server for iSeries” on page 63. Also, refer to *Consolidating Windows 2000 Servers in iSeries: An Implementation Guide for the IBM Integrated xSeries Server for iSeries*, SG24-6056.

2.2.1 Loading files on the deployment server

During the deployment server installation phase, EnterpriseOne software is loaded onto the deployment server by running the setup program that is included on the first CD. The first CD installs release components and the path codes. The second CD installs the help and data files. The third CD installs DB2 UDB Server import files. The fourth CD installs Oracle and the SQL Server import files. The third and fourth CDs also contain the setup scripts for the relational database management system (RDBMS) that are used later in the deployment server installation process to create the databases that are necessary to contain the Central Object tables and other binary large object (BLOB) tables.

Note: In an installation site where iSeries is an enterprise server, you might want to install central objects on the iSeries server, the most common installation scenario. In such a case, choose the Custom option and do not install the DB2 UDB, Oracle, and SQL Server import files automatically from the third and fourth setup CD.

Follow these steps to begin the installation:

1. Sign on the Deployment server as user JDE. The JDE user profile should already belong to the Administrators group.
2. Load the first setup CD. Options are presented, as shown in Figure 2-4.

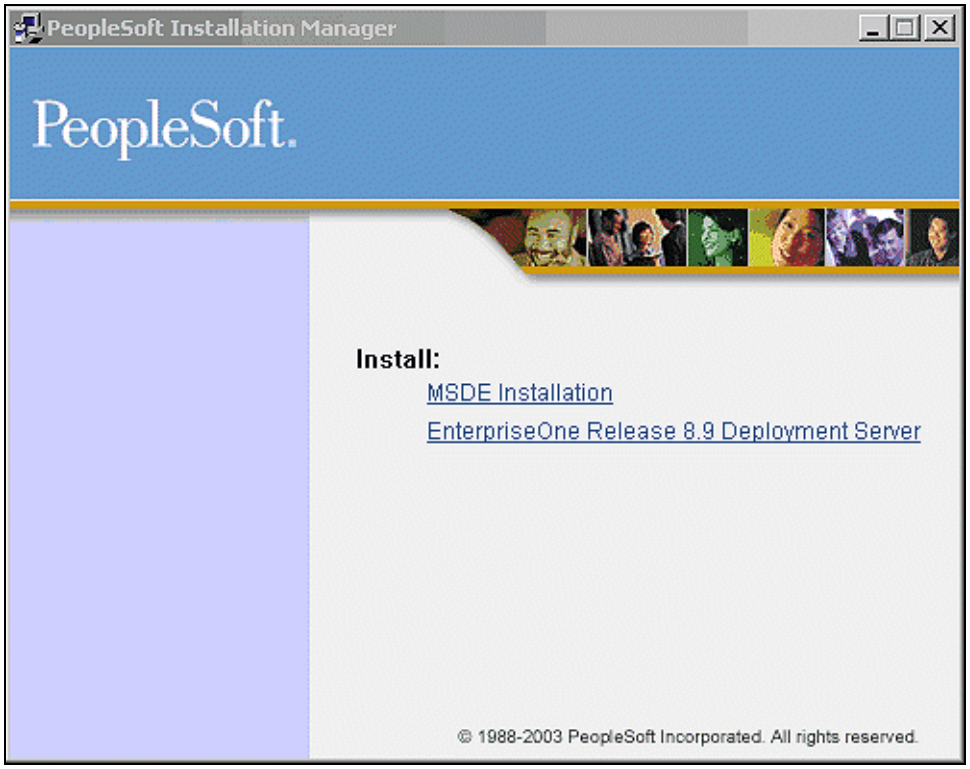

Figure 2-4 Setup type

3. If you do not have Microsoft SQL Server Desktop Engine (MSDE) installed, select **MSDE Installation** to proceed.

> **Note:** You cannot install MSDE from the first deployment server CD if Microsoft SQL Server 2000 Client Tools and Enterprise Manager is already installed on your deployment server.

4. Continue loading EnterpriseOne Release 8.9 deployment server from the first CD. The welcome screen appears as shown in Figure 2-5.

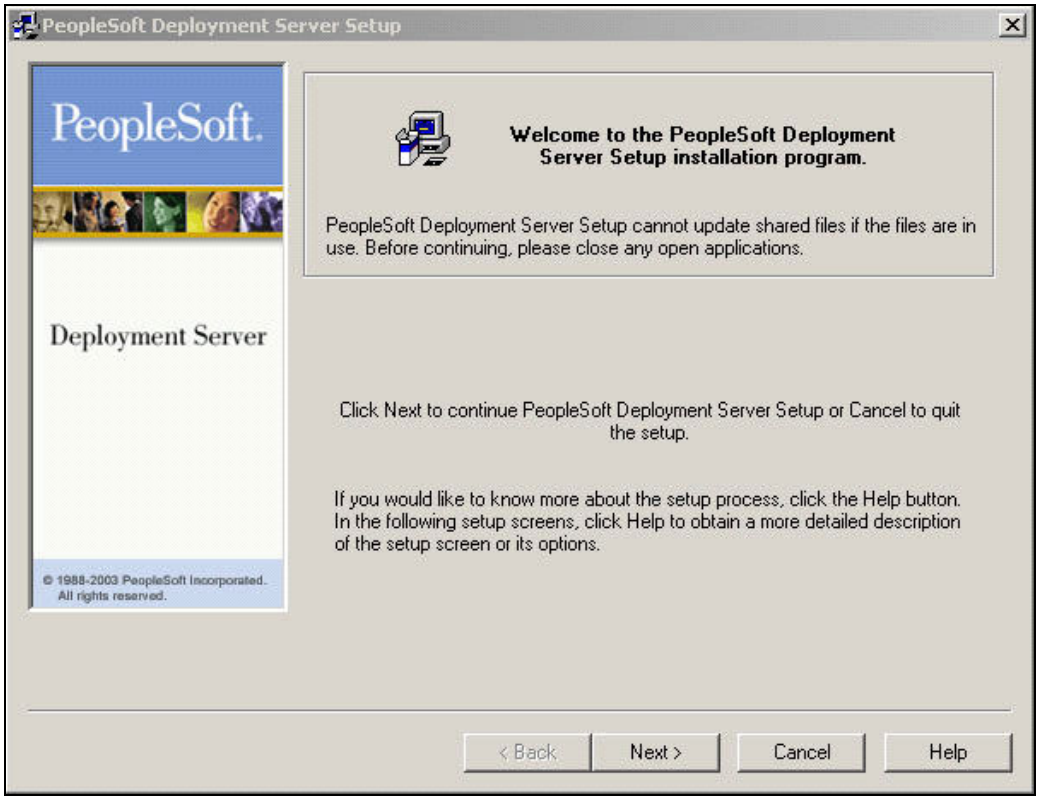

Figure 2-5 Setup Welcome window

5. You must obtain an authorization code from Oracle Global Support Center to proceed further.

Note: During the installation process on the deployment server, if you cancel the EnterpriseOne Installation program before it completes successfully, you will need to obtain a new authorization code.

6. After validating the authorization code, the installation program usually presents the setup options, as shown in Figure 2-6. Select the Custom option to install the system files and the necessary path code objects. You can change the name of the target folder. However, the setup program creates the target folder if it does not exist.

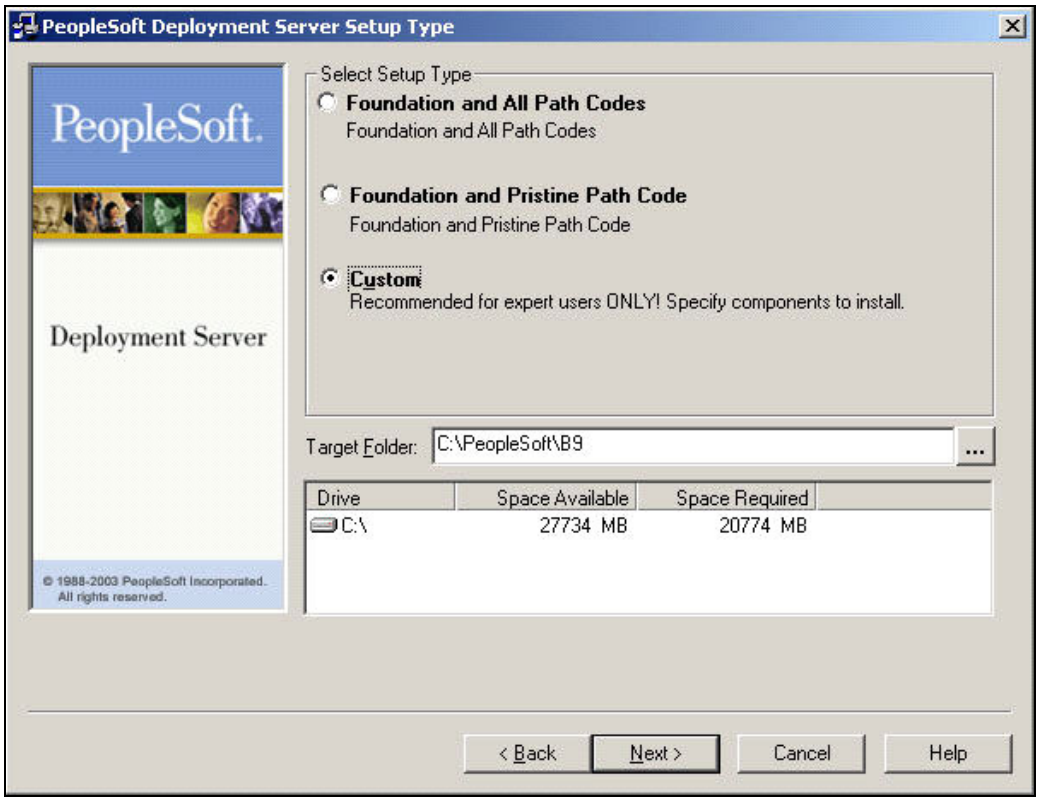

Figure 2-6 Selection of setup type

7. At the Component Selection screen (Figure 2-7), select the path codes that correspond to the EnterpriseOne environments that you plan to install. You must select OneWorld Files, because it installs the system files. Be sure to clear the path code that you do not want to install.

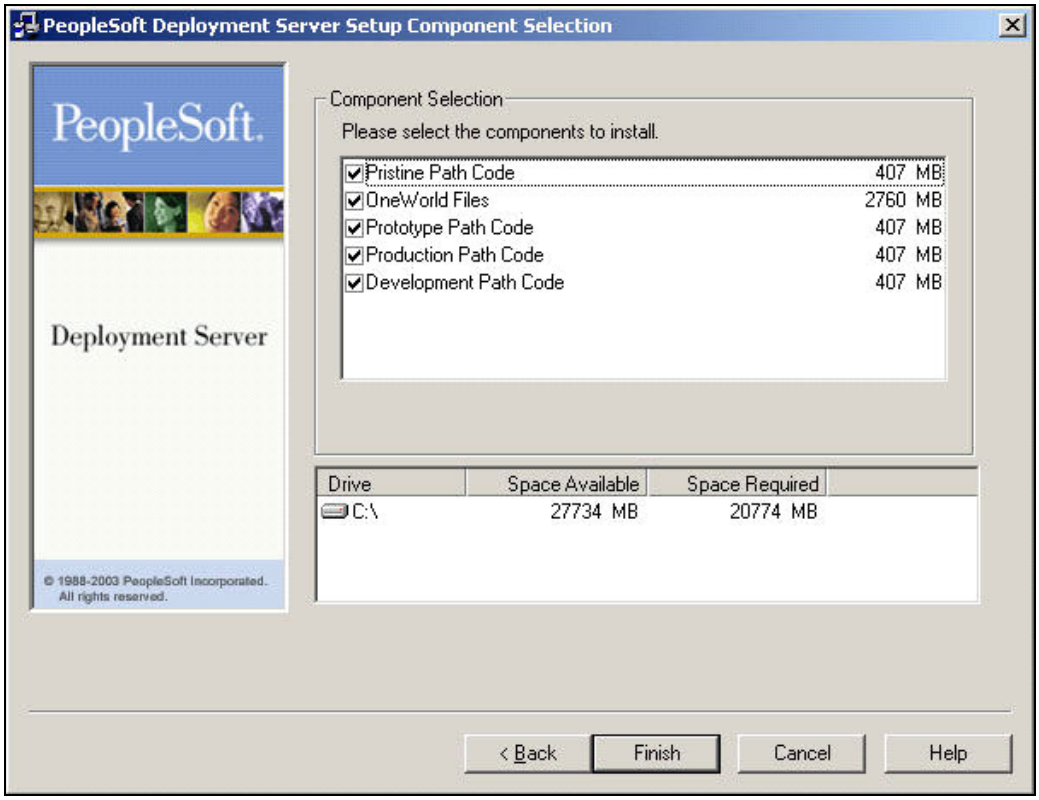

Figure 2-7 Component selection

The deployment server installation progresses as the files are loaded.

8. Load the second CD to populate the help files. The location to where the help files are loaded becomes the default location for all EnterpriseOne help files.

   At the end of the installation, a message appears indicating whether the installation was successful.

2.2.2 Post deployment server installation tasks

After you install the deployment server, you need to:

- Share the directory \PeopleSoft\B9

  You must share the \B9 directory prior to creating the installation plan. Otherwise, the odbcdatasource.inf file is not created. Be sure that the share has full read-write access for the Administrators group.

- Install the latest service pack

  Using the service pack documentation, install the service pack one-off to the deployment server. You need to install the service pack prior to applying the planner update Electronic Software Update (ESU).

- Install the latest planner update

  Install the latest planner update on the deployment server and execute the special instructions as appropriate. You can find information about the latest planner update at the Oracle PeopleSoft Customer Connection at:

  http://www.peoplesoft.com/corp/en/public_index.jsp

2.3 Installing the Enterprise Server

The EnterpriseOne 8.9 software is loaded onto the iSeries server using CD media. There are four CDs referred as host code CDs and three central objects CDs. At the initial stage, the contents of these CDs are copied to a save file library named as OWINSTALL. The contents of this save file are restored during the installation process.

Figure 2-8 illustrates a sample screen shot of the above mentioned OWINSTALL library which can be referred as a temporary storage area for the EnterpriseOne installation process on the iSeries server.

```
                              Display Library

Library  . . . . . :   OWINSTALL       Number of objects  . :   13
Type . . . . . :   PROD            Library ASP number . :   1
Create authority . . :   *SYSVAL         Library ASP device . :   *SYSBAS

Type options, press Enter.
  5=Display full attributes   8=Display service attributes

Opt  Object       *Type     Attribute              Size  Text
     DATADICT     *FILE     SAVF               196755456
     JDEOW        *FILE     SAVF                 2060288
     KRNLSPEC     *FILE     SAVF                68403200
     MODULE       *FILE     SAVF               276684800
     OBJECTS      *FILE     SAVF              1635856384
     SETOWAUT     *FILE     SAVF                 1200128
     SOURCE       *FILE     SAVF               274558976
     SPECS        *FILE     SAVF               793014272
     SRVPGM       *FILE     SAVF               257163264
     SYSTEM       *FILE     SAVF                85495808
     USRSPC       *FILE     SAVF                  155648
                                                                    More...
```

Figure 2-8 DSPSAVF display

2.3.1 Loading files on the Enterprise Server

Follow this process to initiate the process of loading the iSeries installation files:

1. Log on to the iSeries as QSECOFR. Ensure that you are the only user logged on as QSECOFR and that the message queue is not locked by anyone.
2. Enter the command **LODRUN *OPT01***, where *OPT01* is the name of the optical device on the iSeries.

At this stage, the program repeatedly asks for all four host code CDs and then asks for the first volume of the iSeries central objects CDs.

After loading central objects CDs, the library named JDEOW is restored on the Enterprise Server. The Installation Menu (A98OWMNU) appears as shown in Figure 2-9.

```
A98OWMNU                PeopleSoft                          DENPS1
  B9                  OneWorld Installation Menu

 ... DATA LIBRARIES                              ... POST INSTALLATION
  2.  Install Data Dictionary                     9.  Edit INI File (IFS)
  3.  Install Central Objects                    10.  Verify Library List
  4.  Create OneWorld Libraries                  11.  Change SYSTEM Description

 ... SYSTEM LIBRARIES                          ... SERVER CONFIGURATION
  5.  Install OneWorld Server Libraries          12.  Porttest
  6.  Install Additional Path Codes              13.  Start JDE Server
                                                 14.  End JDE Server
                                                 15.  Clear IPC
 ... PRINTER CONFIGURATION
  7.  Create Output Queue                        ... MISC. OPERATIONS
  8.  Create Job Queue                           17.  Display Active Jobs
                                                 18.  Work with Spooled Files

Selection or command


===>
```

Figure 2-9 A98OWMNU display

The following libraries are created:

- SY9
- SV9
- OL9
- DD9
- PRODDTA
- TESTDTA
- CRPDTA
- PRISTDTA
- PRODCTL
- TESTCTL
- CRPCTL

However, most of these libraries are populated during the Installation Workbench.

The user profile JDE is created automatically during this stage with the password JDE. The JDE user profile on iSeries is also used for database authentication during the installation workbench.

2.3.2 Loading central objects on the Enterprise Server

Complete the following steps to load the central objects on the iSeries Enterprise Server:

1. Select option 3 from the installation menu to install central objects, as shown in Figure 2-9.
2. Enter the name of the central objects library that corresponds to the EnterpriseOne environment that you are installing (for example, COPY9, COJD9, COPD9, CODV9). Press Enter.

At the end of the installation process, a message is displayed.

2.4 Installing the deployment server

The EnterpriseOne installation application is used at this stage to define the installation plan through the process of configuring the machines on which you will run Release 8.9. This application is a wizard-like application that allows you to define server types, environments, and data sources for the installation. During this definition process, the application updates the Microsoft SQL Server Desktop Engine (MSDE) planner database (\B9\Planner\Data\JDEPlan.mdf) on the deployment server.

2.4.1 Installation plan

Primarily, two types of installation plans are used in a standard installation scenario:

Typical Assumes all EnterpriseOne defaults through the planning process and thus allows very little flexibility.

Custom Allows for customizations to the defaults, such as to allow custom naming conventions or to choose the environment to be installed.

Oracle recommends using the Custom Plan for any standard install because it offers the flexibility to make any needed changes to the ODBC data sources. After the plan definition, the Custom Plan goes through the finalization and validation process. When the plan is finalized, the plan status is changed to 20. This status change signals that several adjustments to the tables have been made according to the plan.

Some of the tables that are updated are:

- Release Master (F00945)
- Path Code Master (F00942)
- Machine Detail (F9651)
- Package Plan (F98404)

The Custom Plan also creates the also creates the Object Configuration Master (OCM) and modifies the INI file.

During the validation process, the planner validation report (R9840B) is run. This process ensures that the plan contains all of the necessary information. At the end of a successful validation process, the status of the installation plan is set to 30.

In order to start the installation planner in the planner environment (JDEPLAN), log on to EnterpriseOne as user JDE, using the password JDE. The planner environment uses the planner path code on the deployment server.

Note: Do not start the Planner Validation and Workbench prior to loading the host code and central objects CDs that are defined in 2.3.1, “Loading files on the Enterprise Server” on page 14”. Otherwise, the installation will fail because the required libraries do not exist on the enterprise server.

For the detailed step-by-step process on running the installation planner, refer to the “Starting Installation Planner” section in *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

2.4.2 Installation Workbench

The EnterpriseOne Installation Workbench executes the plan that has been validated through a series of workbench processes as described in 2.5.1, “Modifying the Enterprise Server

jde.ini file" on page 18. The process basically copies tables from the planner environment into the data sources of the defined environments on the appropriate servers.

You can run the Installation Workbench either in attended or unattended mode. The Installation Workbench consists of several separate workbench functions that include:

- Initial Tasks Workbench: Performs the Release Master. At this stage, mostly empty System-B9 tables are created and the Release Master table (F00945) is populated.
- Location Workbench: Copies location information defined in the plan from the Deployment Locations Definition table (F9654) in the Planner data source to the System - B9 data source. It also updates the Location Plan Detail table (F984021).
- Data Source Workbench: Copies all data sources defined in the plan from the F98611 Data Source Master and F986115 Data Source Sizing tables in the planner data source into the System - B9 data source. It also updates the Data Source Plan Detail table (F98401) to indicate completion.
- Environment Workbench: Copies the F0094, F00941 and F00942 Environment Information tables, as well as the Object Configuration Master table (F986101) for each environment, from your planner data source to your System - B9 data source. It then updates the Environment Plan Detail table (F98403) to indicate completion. Depending on the plan settings, it runs a batch application (R98403) to create all EnterpriseOne tables in appropriate data sources from JDELocal.mdf and loads demo data as appropriate from JDEData.mdf.
- Machine Workbench: Copies the Server Configuration tables (F9650 and F9651) from the planner data source to the System - B9 data source. It then updates the Host Plan Detail table (F98402) to indicate completion and utilizes the plan environment information to populate the Object Configuration Master (F986101) and the Data Source Master (F98611) tables in the Server Map data source.
- Package Workbench: Copies the Package Information tables (F9603 and F9631) from the planner data source to the System - B9 data source. It also updates the Package Plan Detail table (F98404).

Before beginning the Installation Workbench process, you need to:

- Disable debugging

  It is highly recommended that you disable debugging during the execution of the Installation Workbench. If debugging is on, the performance of the process is degraded, and a large log file is created that can consume disk space.

  To ensure that debugging is off, check the jde.ini file located in the \winnt directory. The OUTPUT statement in the DEBUG section should be set to `NONE`. Make sure no logging level is turned on. If any logging level is turned on, performance is impacted in spite of having set `OUTPUT=NONE`.

  For a list of the contents of the jde.ini file, see Appendix B, "JD Edwards EnterpriseOne 8.9 test configuration files" on page 125.

- Verify the JDE user profile and passwords

  It is a mandatory to have the JDE user profile enabled on your iSeries server. If the JDE user profile does not exist, the Installation Workbench fails. You should also verify that it has the same password as the JDE user profile on the deployment server.

  The passwords of the JDE user profile on the deployment server and the iSeries enterprise server must be synchronized on both systems to allow successful ODBC connections to be established. If the passwords do not match on the two systems, each attempt to initialize an ODBC connection causes a database sign on window to be issued that asks you to enter the correct iSeries server password.

In addition, verify that the iSeries server JDE user profile has *JOBCTL and *ALLOBJ special authority. Apply these special authorities to ensure successful completion of the Installation Workbench.

Furthermore, verify that the *CHANGE object authority is assigned to *PUBLIC for the iSeries server JDE user profile. This assignment is required later for EnterpriseOne client workstation users to submit Universal Batch Engines (UBEs) to run on the iSeries server.

After successful completion of the workbench, apply the Electronic Software Update (ESU) and available Application Software update (ASU).

For a detailed step-by-step process of running the Installation Workbench, refer to the "Running Installation Workbench" section of *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

2.5 Installing the Enterprise Server

During this phase of the installation process, EnterpriseOne server libraries, the pathcode library, foundation library, and associated directories in the Integrated File Systems (IFS) are installed. This phase can be initiated after the Planner Validation workbench is successfully completed. Before beginning the installation, you need to perform the tasks described in this section, as appropriate.

2.5.1 Modifying the Enterprise Server jde.ini file

The jde.ini file, located in the \PeopleSoft\b9\hosts\AS400\ enterpriseservername directory in the deployment server, is transferred during the installation of the server libraries process (C98OWINST). You need to modify this .INI file to remove any unwanted path codes. Update the necessary parameter with a valid environment name as follows:

- Security: The DefaultEnvironment parameter in the [SECURITY] section must match a valid environment on the plan. An example is DV9.
- Remove unwanted path codes: Alter any statements in the jde.ini that file relate to path codes that are not installed to reflect the installed path code values.
- Port number: Ensure that the port number specified in the [JDENET] section of the jde.ini file (for ServiceNameListen and ServiceNameConnect values) matches the port number specified during the Installation Planner process.
- Update Server Name: Change any statements in the jde.ini file relating to the name of the server to a valid server name as appropriate.

For a list of the contents of the jde.ini file, see Appendix B, "JD Edwards EnterpriseOne 8.9 test configuration files" on page 125.

2.5.2 Modifying the library file that resides on the deployment server

The library file located at \PeopleSoft\b9\hosts\AS400\enterpriseservername on the deployment server is used to determine the path codes to be installed on the enterprise server. This file is transferred to the enterprise server at the beginning of the installation of the server libraries (C98OWINST) process. You need to modify this file based on your installation scenario. The first entry of this file should always be B9SYS.

2.5.3 Installing the server libraries

The installation process of the server libraries does not require the CD media. The necessary installation files are loaded on the iSeries server through the process described in 2.3.1, “Loading files on the Enterprise Server” on page 14.

To install the server libraries, you need to:

1. Log on to the iSeries as QSECOFR. Ensure that you are the only user logged on as QSECOFR and that the message queue is not locked by anyone.
2. Enter the command **ADDLIBLE JDEOW** at the command prompt.
3. Enter the command **GO A98OWMNU** at the command prompt.
4. From the Installation Menu (A98OWMNU), choose option 5 to install the Server libraries. The menu appears as shown in Figure 2-10.

```
                    Install OneWorld Libraries (C98OWINST)

Type choices, press Enter.

Save File Library  . . . . . .   OWINSTALL     Character value
Release Number . . . . . . . .   B9
Deployment Server Name . . . . .
OneWorld Deployment Path . . . .   '?:\PeopleSoft\B9'
Deployment Server User ID  . . .   JDE           Character value
Deployment Server Password . . .                 Character value
Change Messages to Break Mode  .   *YES          *YES *NO
```

Figure 2-10 Installing OneWorld Libraries display

5. Enter the necessary parameters, such as the Deployment server name, the password of the user ID for JDE on the deployment server, and the deployment path driver letter where EnterpriseOne is to be installed (replacing the question mark). You can ignore the drive letter if the virtual directory is properly configured in the deployment server's FTP setup.
6. Press Enter to continue the install process.

   At this stage, the installation process submits the J98OW20 job to batch when the library and INI files are transferred to the iSeries from the deployment server. If any problem appears during the file transfer process, use the Display Physical File Member command (**DSPPFM**) to check the FTPOUTPUT file residing in the JDEOW library.
7. Enter the Work with Submitted Jobs command (**WRKSBMJOB**) to monitor the installation process. Locate the J98OW20 job.
8. When the batch job completes, review the job log to verify the installation status. To do so, find the spooled file QPJOBLOG reflecting the installation messages.
9. After the installation, follow the instructions in the service pack installation guide to apply the service pack on the enterprise server. The service pack should be the same as is applied on the deployment server. To ensure that the EnterpriseOne supplied Link Business Function command (**LINKBSFN**) is run successfully as part of the service pack install process, review the section named “Tips for using LINKBSFN when applying the service pack” from J*.D. Edwards OneWorld Xe Implementation on IBM @server iSeries Servers* Redbook, SG24-6529.

2.5.4 Creating an output queue for the iSeries default printer

To create an output queue, you need to:

1. Log onto the iSeries server as QSECOFR.
2. Use option 7 from the Installation Menu (A98OWMNU) to create an output queue. Option 7 presents the menu that is shown in Figure 2-11.

```
                    Create OneWorld Outqueue (C98OWOUT)

Type choices, press Enter.

Outqueue Name . . . . . . . . . Character value
Printer IP Address . . . . . .
Remote Printer Queue . . . . . ' '




Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display F24=More keys
```

Figure 2-11 Creating OneWorld Outqueue display

3. Enter the preferred output queue name, appropriate network address for your printer, and the remote printer queue, if necessary.

> **Note:** The value that you enter for the remote printer queue parameter is specific to the make and model of printer to which the output queue is associated. This information should be available in the printer's user manual or, alternatively, from the manufacturer.

4. After successful creation of the output queue, use the Display Joblog command (**DSPJOBLOG)** from the same session and obtain on which library the output queue is created. This information is needed during the set up process of the EnterpriseOne printer.

2.5.5 Creating the job queue

To create the job queue:

1. Determine an unused sequence number. To do so, log on as QSECOFR and enter **DSPSBSD SBSD(QBATCH)**. Choose option 6 for the “Job Queue Entries” and then record an unused sequence number (for example, Seq Nbr 30) based on a sample result as shown in Figure 2-12.

```
                        Display Job Queue Entries
                                                              System:   DENPS1
Subsystem description:   QBATCH         Status:   ACTIVE

 Seq  Job                        Max   ---------Max by Priority----------
 Nbr  Queue       Library     Active             1   2   3   4   5   6   7   8   9
10  QBATCH        QGPL           1               *   *   *   *   *   *   *   *   *
20  QS36EVOKE     QGP         *NOMAX             *   *   *   *   *   *   *   *   *
50  QTXTSRCH      QGPL        *NOMAX             *   *   *   *   *   *   *   *   *
```

Figure 2-12 Displaying job queue entries

2. Enter the command **ADDLIBLE JDEOW** on the command prompt.
3. Enter the command **GO A98OWMNU**, and choose option 8 to create the job queue. The Create OneWorld Job Queue display appears (Figure 2-13).

```
                    Create OneWorld Job Queue (C98OWJOB)

Type choices, press Enter.

Job queue . . . . . . . . . . . QB9 Character value
Sequence number . . . . . . . . 30
```

Figure 2-13 Creating OneWorld Job Queue on the iSeries enterprise

4. Type the preferred job queue name and sequence number recorded in 2.5.4, “Creating an output queue for the iSeries default printer” on page 20, and press Enter.

2.5.6 Running the PORTTEST program

To verify that EnterpriseOne for a specific environment is installed correctly, run the PORTTEST program after the installation. PORTTEST initializes a user and an environment (assuming that Release 8.9 is installed and configured correctly).

To invoke the PORTTEST program:

1. Log on to the iSeries enterprise server as ONEWOLRD or JDE and then choose the option 12 from the menu A98OWMNU.
2. Enter **JDE** for the user ID, the password for JDE, and the environment to be tested (for example, PY9, DV9, PD9).
3. Press Enter.

Any failures during the processing of the PORTTEST program can suggest a problem with the configuration of the selected OneWorld environment. There are a number of reasons that can cause PORTTEST to fail. A good listing of such possible reasons can be found in the guide *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

2.5.7 Starting EnterpriseOne services

You need to start EnterpriseOne services to ensure that all necessary processes are working as expected. You can start EnterpriseOne services from the command line or from the A98OWMNU menu.

To start from the command prompt:

1. Log on to the iSeries enterprise server as ONEWORLD.
2. Choose the version of the EnterpriseOne that you want to start from the Current OneWorld Versions menu. It automatically places the corresponding foundation library (for example, B9SYS) in the library list for that iSeries session.
3. Enter following commands in order to initiate EnterpriseOne services:

```
CLRIPC

STRNET
```

The **CLRIPC** command clears the IPC™. The **STRNET** command invokes the subsystem (for example, JDEB9) where EnterpriseOne kernel and network processes are run.

When invoking **STRNET**, verify that there is a NETWORK entry running with a program name of PGM-JDENET_N in SELW status. This NETWORK is the primary JDENET communication job. According to .INI file settings, additional JDENET_N jobs are started. In addition JDENET_N jobs start JDENET_K ("Kernel" jobs).

Note: The SENTINEL process is no longer used in EnterpriseOne 8.9.

Alternatively, you can start EnterpriseOne services using the A98OWMNU menu. In this case, you need to clear the IPC from option 15 and then start the Enterprise One services using menu option 13. Again, you can use option 17 to verify the expected primary processes resulting from the services.

EnterpriseOne services must be active before you can run the PORTTEST program, if the OneWorld Security Server or JDBNET processes are configured for use to communicate with other server.

2.5.8 Setting up an EnterpriseOne printer

When the EnterpriseOne installation is successfully completed, you need to configure EnterpriseOne printers using an EnterpriseOne application. You configure a printer through the Printers Menu (GH9013).

Note: During the process for setting up a printer, you need to provide the printer name. This name must be the same as the output queue name. In this case, enter the name of the output queue for the output queue name that corresponds to the output queue that you created in 2.5.4, "Creating an output queue for the iSeries default printer" on page 20. You also need the name of the library where the output queue resides.

For details on the process of setting up an EnterpriseOne printer, refer to the guide *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

2.6 Installing client workstations

In this phase, EnterpriseOne software is deployed from the deployment server to client workstations. This installation process is done using either interactive mode, silent mode, or push mode.

At the initial stage, to test the installation, usually interactive mode is used to install client workstations. Push mode is described in the *Package Management Guide*. (You can access this *PeopleBook* via the Oracle PeopleSoft Customer Connection with an authorized login. Select **Support** → **Documentation** → **Documentation updates**.)

You must also ensure that the client workstations have the most up-to-date iSeries Access Service pack, as recommended in the IBM-JDE APAR Web site:

http://www-1.ibm.com/servers/enable/site/events/jde_links.html

If the iSeries Access service pack is not the same level between the deployment server and the client workstations, the installation can fail. This client installation failure can only happen if the location of the cwbodbc.dll file between the deployment server and the client workstations is different, and thus, the necessary ODBC data sources are not created.

The client workstation installation process produces a log file that is named jdeinst.log. You should review this file to identify the installation status.

To install a client workstation either in interactive or silent mode, refer to the “Installing Release 8.9 on the Workstations” section of *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

2.7 Completing the install

After the main installation process is complete and prior to using EnterpriseOne 8.9, you should do the following based on the configuration scenario:

- Modify the media object queue to point to the correct location where the actual image files or OLE objects should reside.
- Set up the solution explorer.
- Execute other application related tasks as appropriate.

For a detailed step-by-step process on these processes, refer to the “Completing the Installation” section of *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System* .

3

JD Edwards EnterpriseOne 8.9 performance management

The key to good system performance remains the same regardless of the applications that are running on the system. The key to good system performance is to provide sufficient system resources in a balanced configuration. A well performing system has the available processor, memory, disk, and network capacity, individually and in balance with one another, that are sufficient to handle the application workload so that no performance bottlenecks exist. Performance management, the technique used to make sure that a system performs at its best, is an ongoing, cyclical process and consists of three key steps:

1. Measure and analyze
2. Adjust
3. Re-measure

For optimal system performance, you should repeat these steps on a regular basis. Results can also be used to predict and schedule required system upgrades as well as to resolve specific performance problems.

It is important to tune parameters on the client and server for the best performance of the EnterpriseOne Web client environment on the IBM iSeries platform. However, there is little to tune on the client side, because the user interface is a Web browser.

On the server side, you can tune the following aspects:

- Apache HTTP server
- WebSphere Application server
- EnterpriseOne (the application)
- DB2 UDB for iSeries (the database)

This chapter focuses how to examine and optimize the performance on the server side of the Web client environment that is running EnterpriseOne 8.9 with IBM WebSphere Application Server 5.0.

For a list of the specific software and hardware levels used in the test environment, see Appendix A, “The JD Edwards EnterpriseOne 8.9 test environment and transaction details” on page 119.

© Copyright IBM Corp. 2005. All rights reserved.

3.1 Tuning Apache HTTP server performance

The IBM HTTP server for iSeries (5722-DG1) product offers two HTTP servers.

- HTTP server (original)
- HTTP server (powered by Apache)

For WebSphere 5.0 running the EnterpriseOne 8.9 Web client solution, JD Edwards requires the HTTP server (powered by Apache).

You can start, analyze, and tune the Apache HTTP server using iSeries system commands or through a browser interface called the HTTP server administration interface. The browser interface offers ease-of-use features, such as selecting options from lists rather than typing in entries to the configuration file. It also offers extensive help text. The browser interface is the recommended interface for tuning.

Use the following iSeries command to start the browser:

```
STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
```

To access the browser interface, start a Web browser, such as Microsoft Internet Explorer®, and enter the following URL:

```
http://<server-name>:2001
```

In this URL, *<server-name>* is the name of the iSeries server. When prompted to sign on to the server, enter a valid iSeries user profile and password.

You can make configuration changes through the HTTP server administration interface or by directly editing the HTTP configuration file (using the **edtf** command). There is a configuration file for each HTTP server instance located in the IFS directory /www/*<http-server-name>*/conf, where *<http-server-name>* is the name of the Apache HTTP server instance. This directory contains a file, httpd.conf, which holds all of the configuration information for that instance.

The following sections describe how to make changes to the configuration through the browser interface and also list the Apache configuration keywords and values that you can enter directly into the httpd.conf file.

For a list of the contents of the httpd.conf file, see Appendix B, “JD Edwards EnterpriseOne 8.9 test configuration files” on page 125.

3.1.1 Measuring and analyzing performance

To measure the performance of the HTTP server (powered by Apache), use the general iSeries system performance tools, such as Performance Monitor, which is included with the Operating System (OS/400) and Performance Explorer (part of IBM Performance Tools for iSeries, 5722-PT1).

Examine the access log to closely monitor what is happening in the Apache HTTP server instance. By default, the Apache server instance is enabled for access logging. Because the access log impacts performance, one of the performance recommendations is to disable this logging function. Although access logging is disabled, you can re-enable it when there is a need to debug problems.

The Apache server on iSeries provides real-time statistics that you can use to examine what is happening while the Web server is active. Select the **Real Time Server Statistics** link on the left-hand pane to view these statistics, as shown in Figure 3-1.

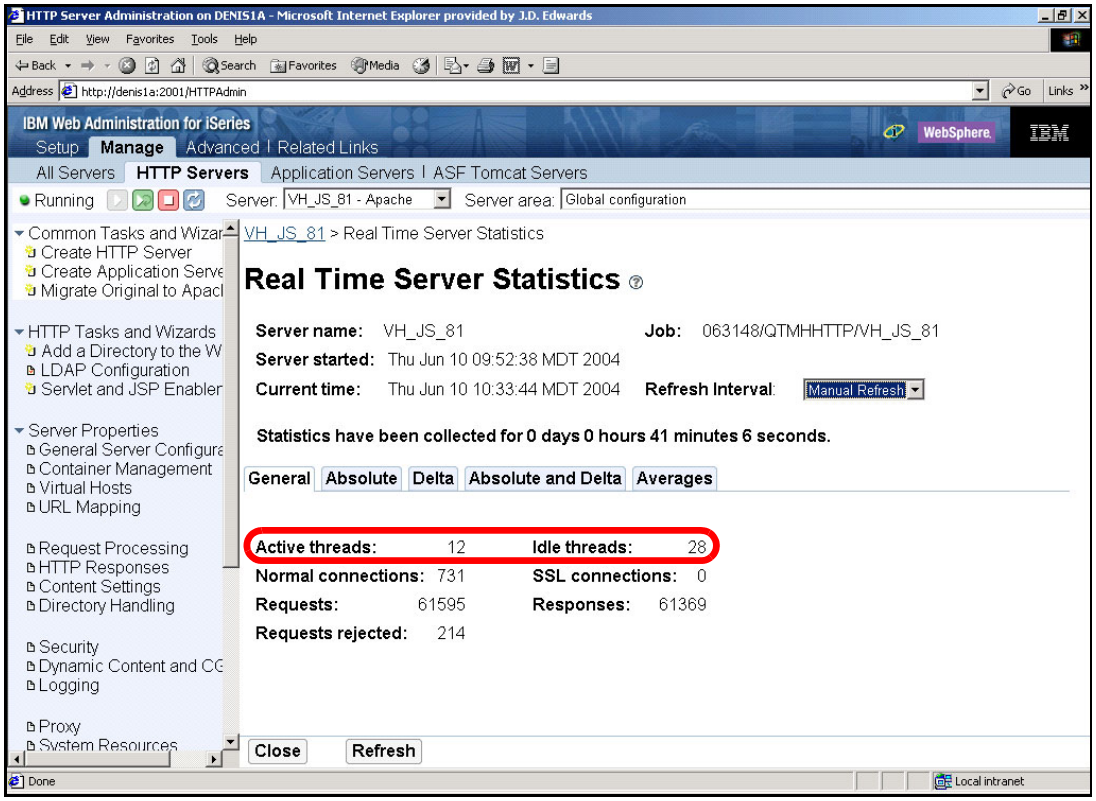

Figure 3-1 Real time server statistics show current activity on the Web server

The server statistics screen shows information about the number of connections in use and the number of requests and responses. Monitor the number of active and idle threads during the time that the system is running its peak load. Adjust the number of threads accordingly. Additional information is available on the Absolute and Delta tabs, such as the byte counts that are sent and received. Use this information to look at the communications workload running on the system during peak time periods.

3.1.2 Adjusting settings to improve performance

The Apache server has several settings that you can change to improve the performance of the EnterpriseOne Web client solution. Enable these settings one at a time to ensure that each recommended change provides a benefit.

Changes can be made to the Apache configuration in two ways:

- Select the Web server instance through the Web interface.
- Use the **`edtf`** command to edit the httpd.conf configuration file, which is located in the IFS directory.

Select the Apache HTTP server instance from the list to make the changes through the Web interface, as shown in Figure 3-2.

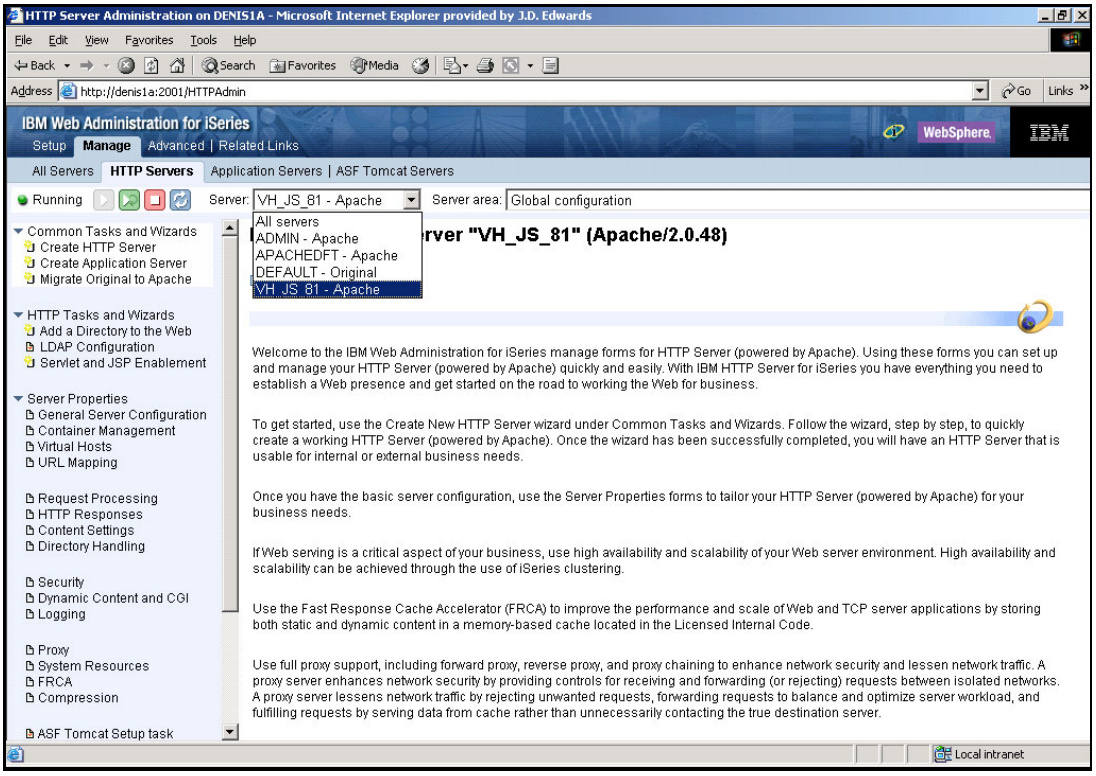

Figure 3-2 Selecting the Apache instance

Make the following changes to the settings associated with this instance:

- Minimize the logging in the production environment

  As discussed earlier, logging is useful during the initial start up of the Web client environment and to debug problems. However, when the production environment is running and stable, disabling extraneous logging provides a performance benefit.

  The Apache HTTP server provides two different types of logging: error logging and access logging. The error log can be configured to capture various types of errors. By default it captures all warnings, errors, and critical conditions. The recommendation is to change this setting so that only errors and critical conditions are captured in the error log. Also, because a separate error log is created for each day that the HTTP server is running, change the configuration to have logs removed automatically after seven days. This setting reduces the disk requirements for the HTTP server.

  To configure the error log:

  a. Select **Logging** from the left pane on the Apache administrative server to change the log settings.

  b. Select **Error Logs**.

  c. Change the Expiration value to 7 days (Figure 3-3 on page 29).

  d. Select the **Logging Level** pull-down list, and select **Error** (instead of Warning).

  e. Click **Apply** after making the change, otherwise the changes are not saved.

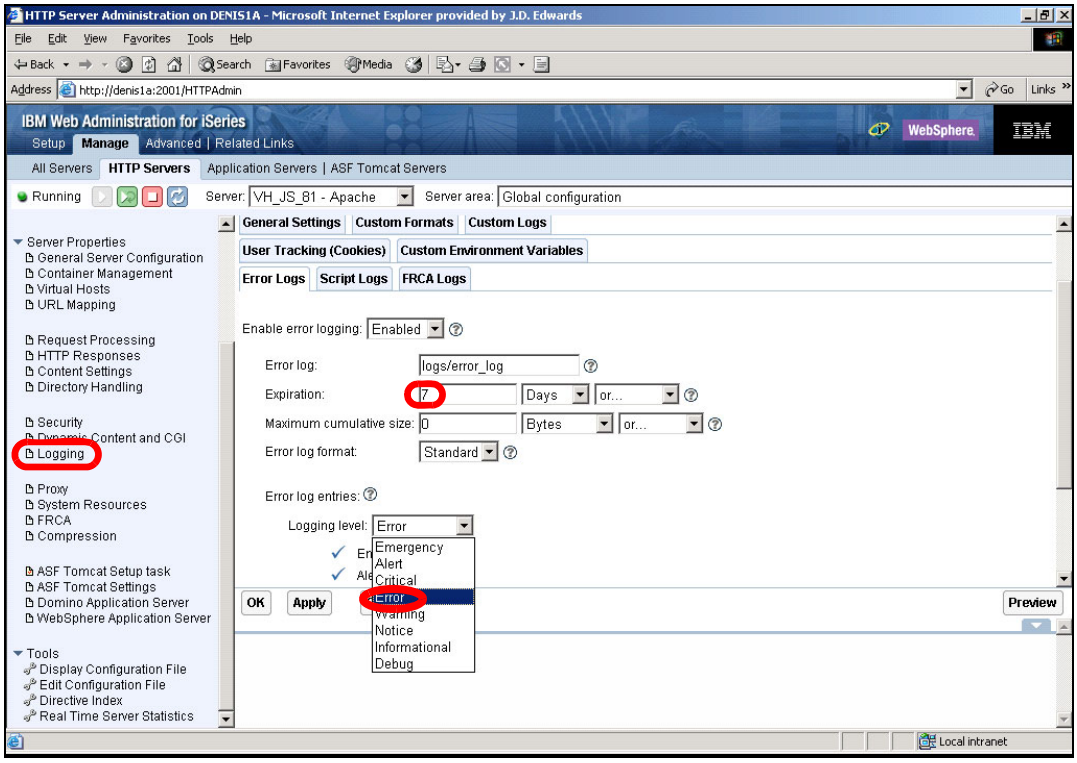

Figure 3-3 Changing the error log settings in Apache

Alternatively, edit the configuration file to add the following two lines:

```
LogLevel Error
LogMaint logs/error_log 7 0
```

The access log, as the name indicates, captures all HTTP accesses to the system. Each browser that accesses the system via a URL is noted by an entry in the access log. Depending on the number of users connected to the system, this log can become very large. As a result, the performance recommendation is to turn off all HTTP access logging.

However, the access log tracks all access to the system, and this log can be used to monitor the external systems accessing the system through the HTTP server or to determine the location of a Web attack. So, for security reasons, you might determine the need to accept any negative performance affect that results from keeping the HTTP access log active on your system.

One way to disable the access log is to select the Custom Format tab on the Logging page.

a. Select each of the log formats and then click **Remove** (Figure 3-4 on page 30).
b. After all of the log formats are removed, click **Apply**.
c. Select the Custom Logs tab, and select the entry for logs/access_log.
d. Select **Remove**, and then click **Apply**.

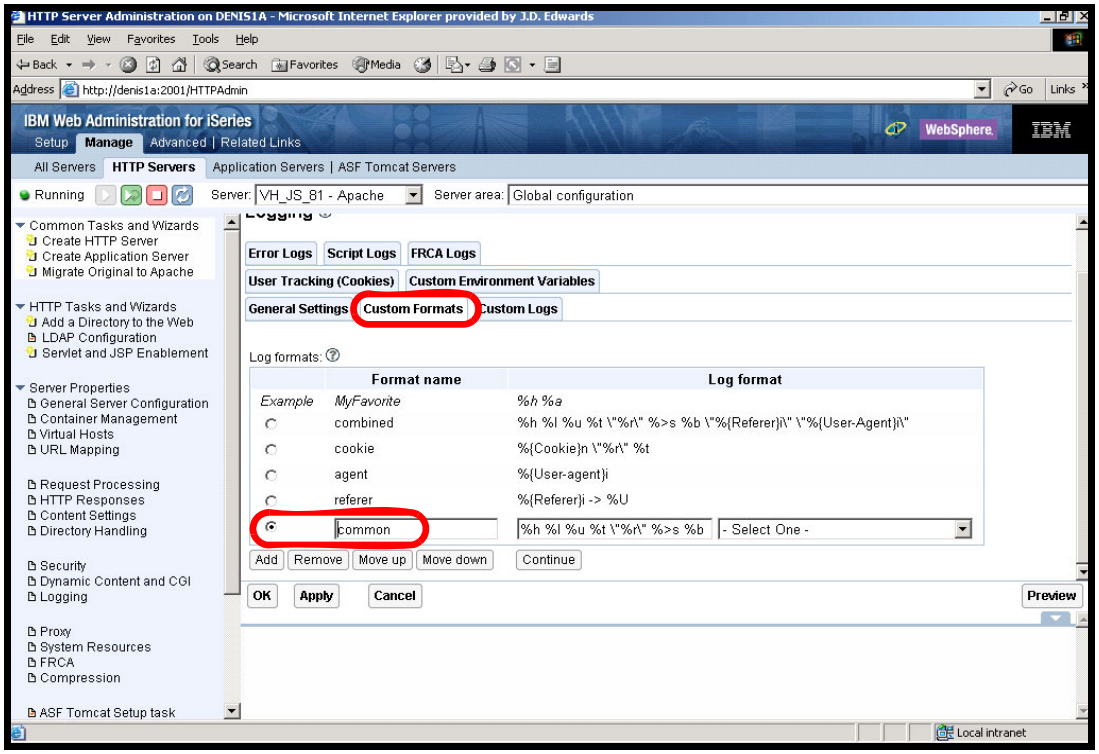

Figure 3-4 Eliminating access logging for the Apache HTTP server Instance

Another method of disabling the access log is to change the httpd.conf file for this instance. Comment out the following lines in the file by putting a number sign (#) in front of each line:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%{Cookie}n \"%r\" %t" cookie
LogFormat "%{User-agent}i" agent
LogFormat "%{Referer}i -> %U" referer
LogFormat "%h %l %u %t \"%r\" %>s %b" common
CustomLog logs/access_log combined
```

If the system has been running with access logging enabled, deleting the existing logs also cleans up system storage. The logs are located in the /www/<http-server-name>/logs directory. Delete all of the files in this directory. The Apache server automatically creates new files as needed.

- Unlimited requests per connection

When a browser on a client machine makes a connection to the Apache HTTP server, the server maintains (or keeps alive) the connection. This allows for multiple requests to be made on the same connection. It is more efficient to keep the same TCP connection for as many requests as possible. By default, the HTTP server uses the same TCP connection for 100 requests. The recommendation is to change the HTTP server setting so that an unlimited number of requests can be sent on the same connection.

  - One method of changing this setting is to select **System Resources** from the left pane of the Apache administrative server and select the HTTP Connections tab, as shown in Figure 3-5 on page 31. Change the number in the *Maximum requests per connection* field from 100 to 0. Though this might seem counter-intuitive, zero indicates that the server allows an unlimited number of requests.

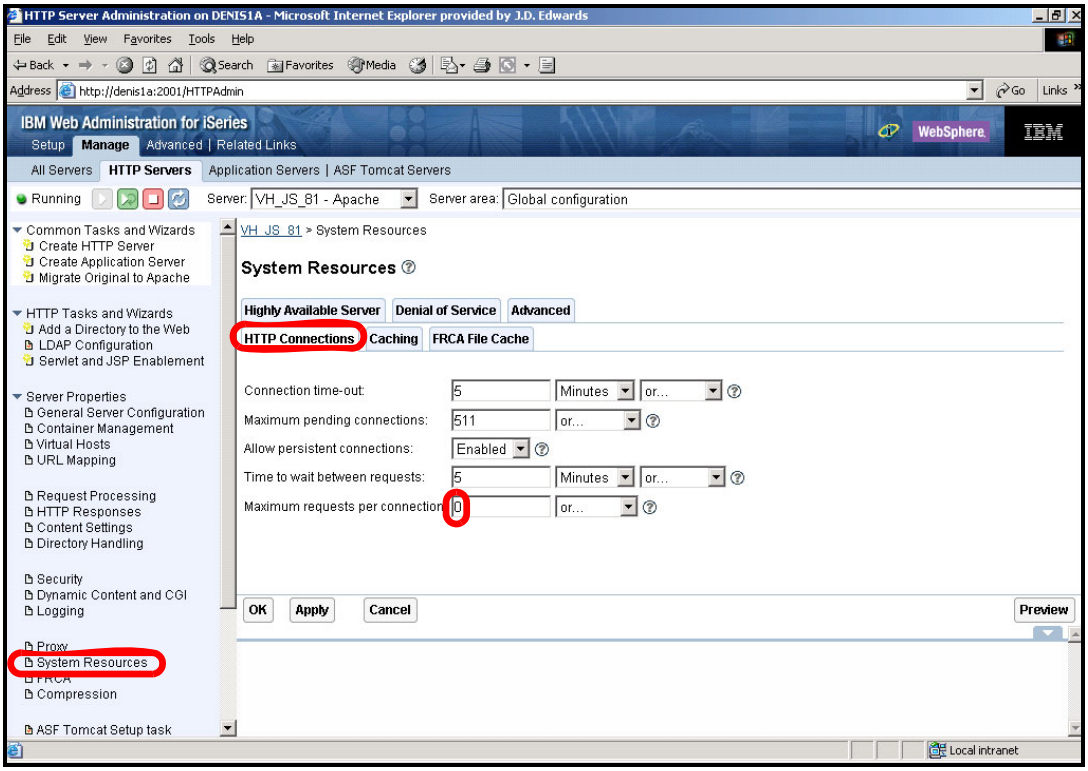

Figure 3-5 Allowing unlimited requests per connection

- Another way to change this setting is to edit the httpd.conf configuration file by adding the following lines:

  ```
  MaxKeepAliveRequests 0
  KeepAlive On
  ```

- Set the number of threads for each job

  The Apache HTTP server job maintains a pool of threads. The number of threads is determined by a configuration setting. The thread pool is used to service all the Web browsers that connect to the server job.

  If all threads within the Web server job are in use and a new request arrives, an additional job is started with the same number of threads in its pool. The recommendation is to keep the proper number of threads available to avoid the extra overhead of starting another Web server job.

  To adjust the number of threads that run in each of these Web server jobs:

  a. Select **General Server Configuration** from the navigation pane on the left-hand side.

  b. Select the General Settings tab.

  c. Change the value in the *Number of threads to process requests* field and click **Apply**.

     The default for this value is 40. Lab tests have found that this number of threads is sufficient to adequately handle 400 users doing typical workloads through browser connections. If the system is running fewer or more users than this, you should change this setting accordingly. For example, if the system is configured for 500 users, change the setting to 50, as shown in Figure 3-6 on page 32.

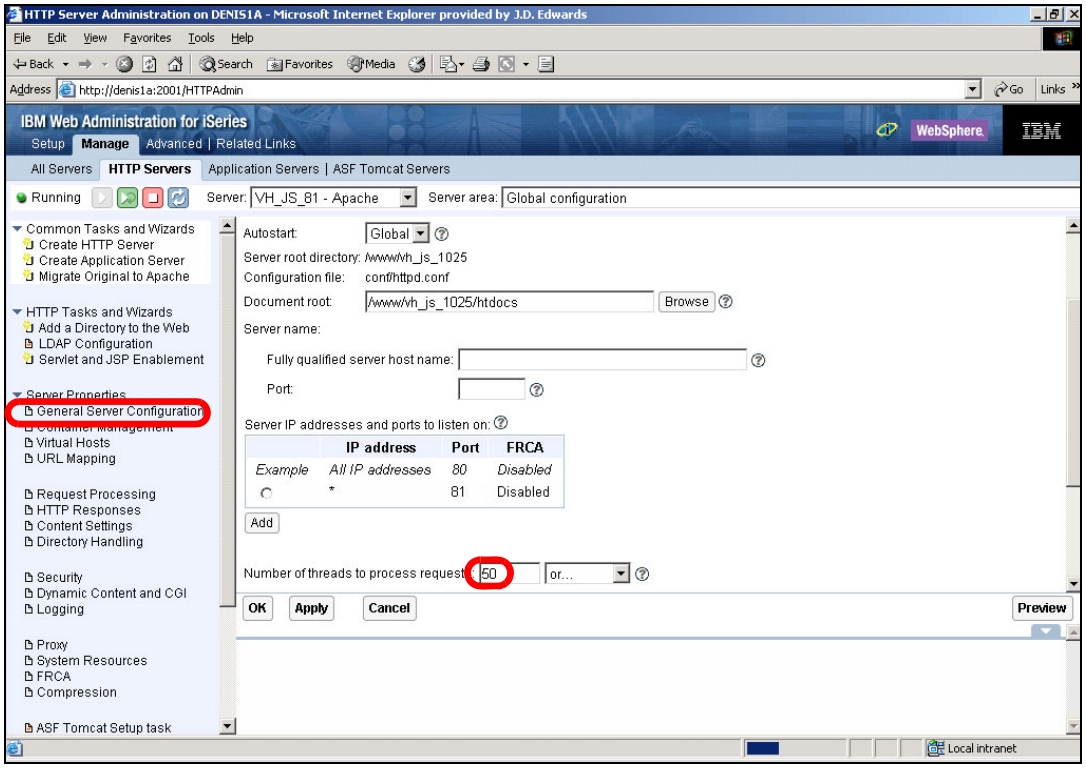

Figure 3-6 Setting the number of threads to handle requests in the Apache HTTP server instance

Alternatively, adjust the number of threads running in each Web server job by editing the httpd.conf file to add the following line:

```
ThreadsPerChild 50
```

- Enable compression

Apache supports HTTP server data compression before files are output to clients that support decompression. This decreases the data volume sent over the network and improves response times. Compression is enabled via a DEFLATE filter, which can be configured to only compress files located in certain directories. Remote clients connected to the network at slow speeds have the greatest potential for performance gains, though locally connected clients also see lesser performance gains.

The Web client's browser version must support data decompression. Internet Explorer and Netscape Navigator both support decompression when running at the versions recommended by the JD Edwards minimum technical requirement (MTR).

To enable compression for the Apache HTTP server, first ensure that the Apache HTTP server is at the correct version. Use the Work with PTF Group command (**WRKPTFGRP**) to ensure that OS/400 V5R2 group PTF SF99098 is at level 13 or higher. The compression function is enabled with level 10. However, the GUI interface to configure the function is enabled with level 13.

To enable compression:

a. Select **Compression** from the left pane (as shown in Figure 3-7).

b. Click the Output Filters tab.

c. In the *Set output filter* section, click **Add**.

d. Enter `DEFLATE` as the filter name, and click **Apply**.

   Alternatively, edit the configuration file by adding the following line:

```
LoadModule deflate_module /QSYS.LIB/QHTTPSVR.LIB/QZSRCORE.SRVPGM SetOutputFilter
DEFLATE
```

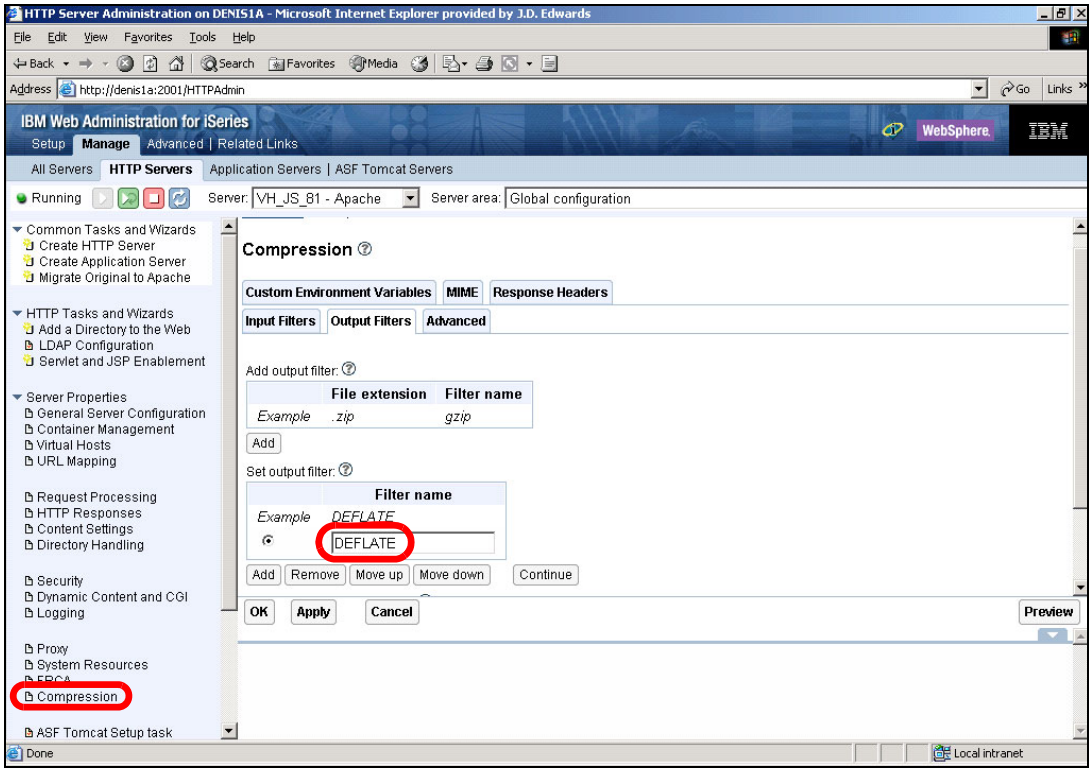

Figure 3-7 Enable compression for the Apache HTTP server

e. In the *Server area* section, select the directory path that ends with webclient.war that was added during the EnterpriseOne Java Application server configuration (Figure 3-8 on page 34).

f. Click **Add** in the *Add output filter by MIME type* section.

g. Enter a MIME type of `text/html` or select it from the list, then enter a filter name of `DEFLATE`. Click **Apply**.

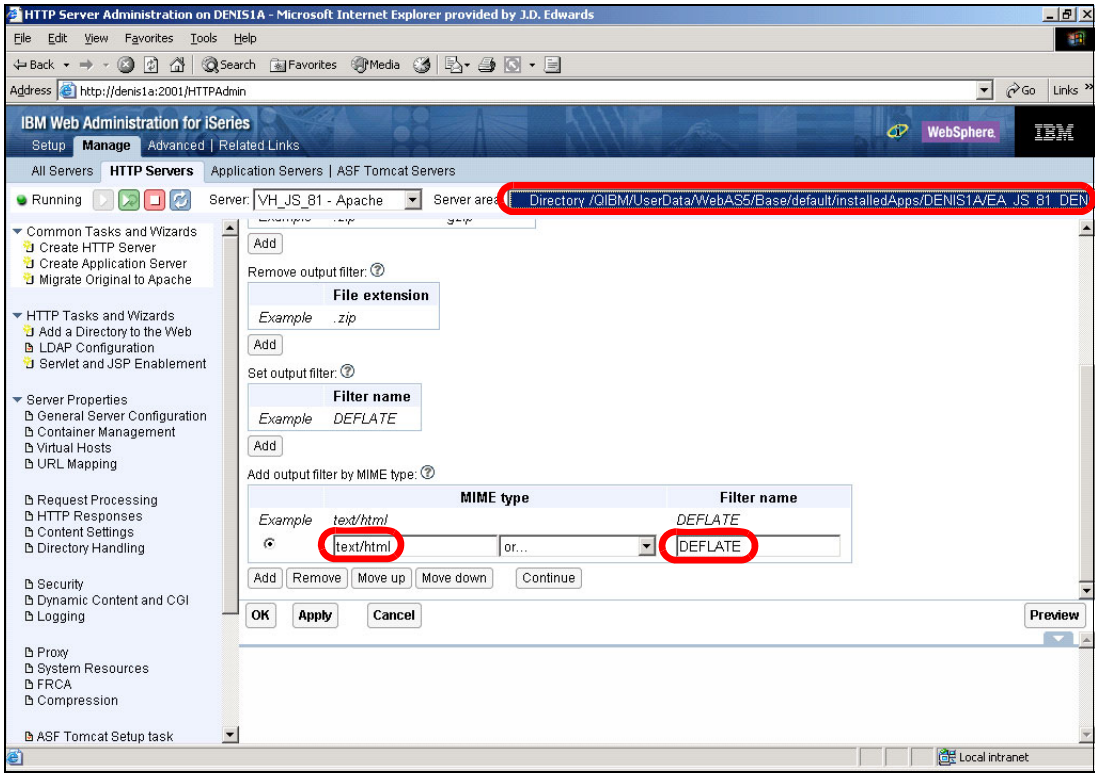

Figure 3-8 Setting the output filter for text/html MIME type in the JAS directories

Alternatively, add a line to the httpd.conf file to change the directory directives to include the `AddOutputFilterByType` statement for text, as shown in the following statement:

```
<Directory
/QIBM/UserData/WebAS5/Base/default/installedApps/<sys>/<ent-app>/webclient.war>
   Order Deny,Allow
   Allow From all
   AddOutputFilterByType DEFLATE text/html
</Directory>
```

In this example, *<sys>* is the name of the iSeries server, and *<ent-app>* is the name of the enterprise application.

3.2 Tuning WebSphere Application Server performance

The WebSphere Application Server 5.0 (5722-WS5) product is the recommended application server for running the EnterpriseOne 8.9 Java Application Server (referred to as JAS). This can be a bit confusing because both products have the term *Application Server* in the name.

To clarify, the WebSphere Application Server is a general-purpose application server that is capable of running many different types of applications. The JD Edwards JAS code is only one of the applications that runs on a WebSphere Application Server. JAS is a set of Java servlets and JavaServer™ Pages™ (JSPs) that are used to serve the EnterpriseOne application interfaces to Web users.

JAS runs with either WebSphere Application Server or WebSphere Application Server Network Deployment. iSeries servers also support WebSphere Application Server - Express. However, JAS does not support this version of WebSphere.

3.2.1 Measuring and analyzing application and JAS servers

You can analyze the WebSphere Application Server, in general, and the JAS server code, specifically, in a variety of ways. You can use the familiar iSeries tools, such as Performance Monitor and Performance Explorer, to analyze WebSphere performance. Additionally, OS/400 has system commands for looking at the Java Virtual Machine (JVM™) which is the WebSphere job that runs the JAS server code.

You can use the Analyze JVM (**ANZJVM**) and Dump JVM (**DMPJVM**) commands to look at heap usage, garbage collection, thread stacks, and object statistics. JAS has a built-in tool, the System Administration Workbench (SAW). WebSphere has a built-in tool, the Tivoli® Performance Viewer. The Tivoli Performance Viewer is also a part of WebSphere 5.0 (5733-WS5). You can use each of these tools to analyze specific internal resources, such as business function response times for JAS and servlet and thread statistics for WebSphere.

The System Administration Workbench is accessible via the following URL:

`http://<sys>:<port>/jde/saw/sawJas.html`

In this URL, <*sys*> is the name of the JAS server, and <*port*> is the port on which the JAS server is listening. You can find additional details on the SAW functionality in 3.4, “Tuning iSeries database performance” on page 44.

The Tivoli Performance Viewer is a useful tool for monitoring resources such as the number of threads used and the amount of memory consumed. A convenient graphical user interface (GUI) monitors WebSphere resource usage and information so that you can adjust the thread settings (Figure 3-9 on page 36).

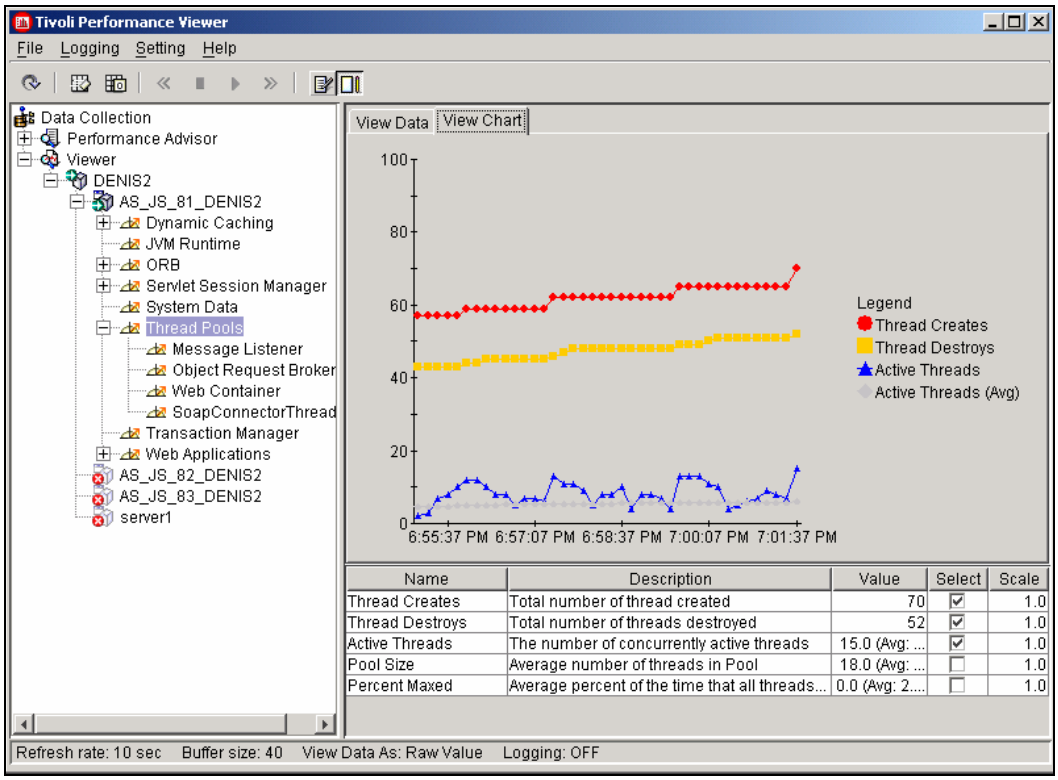

Figure 3-9 Tivoli performance viewer

Although WebSphere 5.0 has replaced the thick client administrative console of previous WebSphere releases with a Web interface, the Tivoli Performance Viewer requires a client. To install the client, run the WebSphere Application Server installation process on the target PC, choose a custom install, and select only the performance monitoring tools.

To run the performance viewer tool, determine which port the application server for JAS is “listening” for RMI connections. The ports for the application are found by following these steps in the WebSphere Administrative Console:

1. Expand the servers.
2. Select **Application Servers**.
3. Click **Your Server Name** → **End Points**.
4. Select the BOOTSTRAP_ADDRESS.

Note the port that is listed. This is the port to use when starting Tivoli Performance Viewer. From a Windows command line, run the following command:

```
tperfviewer <sys> <port> RMI
```

In this command, <*sys*> is the name of the iSeries server, and <*port*> is the Remote Machine Interface (RMI) port for the application server.

When a message is displayed indicating that other application servers are not listening on the specified port, click **OK**. While users are on the system, the viewer displays current statistics on resource usage within WebSphere, and the Performance Advisor provides suggestions on other WebSphere settings to change. The Performance Advisor might provide specific environmental suggestions which should be tested in addition to the tips described in 3.2.2, “Adjusting settings” on page 37.

3.2.2 Adjusting settings

You make changes to the WebSphere 5.0 configuration through the WebSphere Administrative Console. The WebSphere 5.0 Administrative Console uses a Web interface. The default instance of WebSphere is accessed through the following URL:

```
http://<sys>:9090/admin
```

In this URL, *<sys>* is the name of the iSeries server that is running WebSphere.

The administrative console presents a user ID prompt which is used to track the changes entered into the administrative console. It does not have to match an iSeries user profile.

You can tune the following properties within the WebSphere Application Server:

- Adjust the heap size for the JVM

  On iSeries servers, Java performs memory management for applications running within the JVM. It allocates memory on behalf of the application in a single heap space and cleans up unused memory via the garbage collection process. As a Java application, WebSphere has its memory controlled by the JVM. When the application server running JAS is started through WebSphere, Java allocates memory and manages it on behalf of JAS. Setting the initial heap size for the JVM controls the amount of memory used by WebSphere.

  Because Java objects are cleaned up by the JVM, the iSeries JVM also uses the initial heap size as the indicator for when to perform a garbage collection. Therefore, setting the initial heap size to an appropriate value for the workload is very important. Setting the heap size too small causes frequent garbage collections, which consumes processor resources. Setting the heap size too large causes excessive memory usage, paging, and can result in a higher cache miss ratio because the heap is so large.

  The initial heap size setting for JAS is based on the number of active users. The guideline derived from lab tests is 3.3 - 5.0 megabytes per active user. The active user count is defined as the number of concurrent users who are hitting `OK` or function keys two to three times per minute. The appropriate setting can vary depending on the type of application being run as well as the number of users.

  The maximum heap size controls how large the JVM heap is allowed to grow. The recommendation is to leave this field blank, meaning the system controls the maximum heap size. In certain cases on memory-constrained systems, setting the maximum heap size can help the overall system performance.

  To change the JVM settings for the application server,:

  a. Expand the servers.
  b. Select **Application Servers**.
  c. Click **Your Server Name** → **Process Definition** → **JVM Settings**.

  Changing the JVM settings for a 200-user workload is shown in Figure 3-10 on page 38. In this example, you can see that the initial heap size is set to 200 users times 3.3 MB (which equals 660 MB). A maximum heap size has not been set.

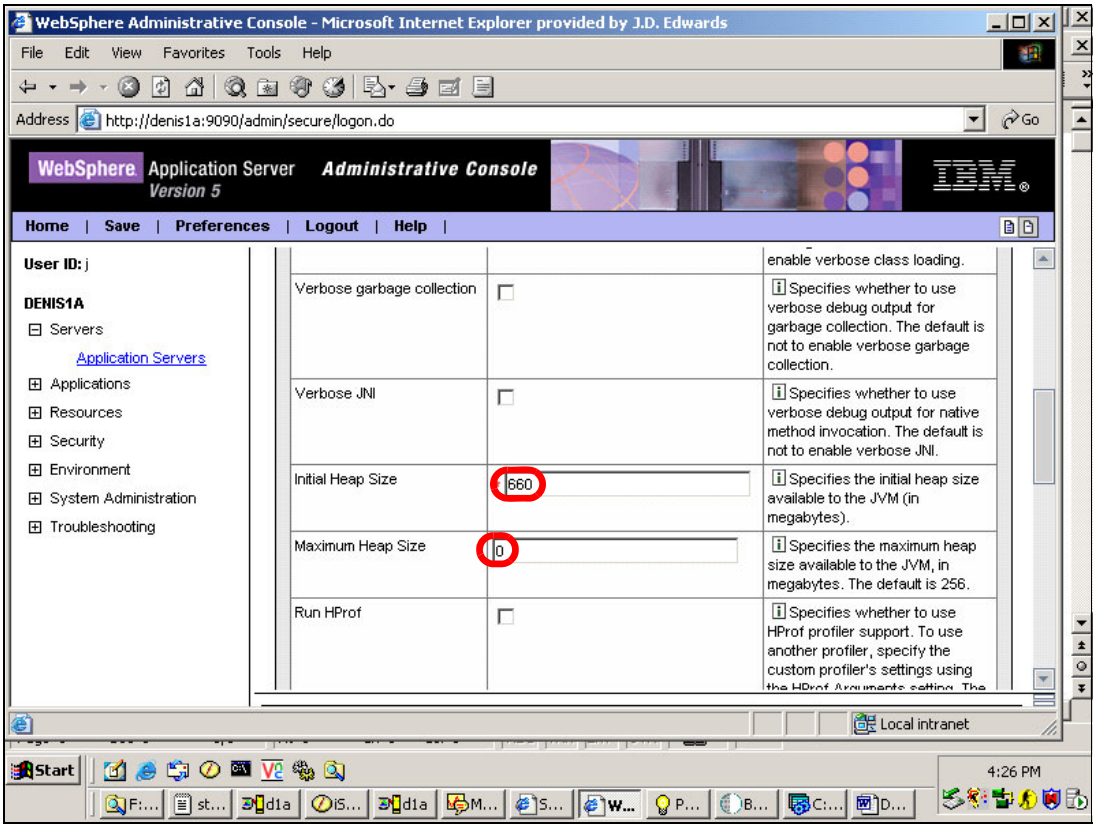

Figure 3-10 Set the initial heap size for WebSphere

You can check heap size and garbage collection information with the Dump JVM (**DMPJVM**) command. This command produces a spool file that contains information about the current heap size, the number of garbage collections performed, and the duration of the last garbage collection.

Note: The WebSphere administrative server restarts any application server job that fails automatically. This is accomplished by periodically checking the job (**ping**), and waiting for a reply. If the iSeries server is underpowered or heavily loaded, the application server job might not respond in time, causing the administrative server to inadvertently end and restart the application server job.

To prevent this from happening, do the following:

a. Expand the servers.
b. Select **Application Servers**.
c. Click **Your Server Name** → **Process Definition** → **Monitoring Policy**.
d. Increase the “Ping timeout” value from the default of 60 to 300.

- Adjust WebSphere subsystem settings

  As shipped, the WebSphere subsystem runs in the base memory pool on the system. In many cases, it is helpful to divide workloads into separate shared memory pools. Putting WebSphere into a separate shared memory pool isolates it from other workloads and ensures that it does not compete with other jobs for memory. Using a shared memory pool also provides flexibility. You can move memory easily and dynamically from one pool to another automatically using the OS/400 command line or a scheduled command.

Note: Under certain circumstances, such as when limited memory is available for WebSphere with multiple JVMs configured, you can achieve better performance if you keep all the memory in the *base pool (as shipped) and let the system manage memory.

To set up WebSphere to run in a separate memory pool:

a. Set up the shared memory pool.

   While shared memory pools already exist on the system, no memory is allocated to them. The shared memory pool for WebSphere running JAS with 200 users requires about three times the initial heap size (660 MB) or 2 GB of memory. Also, the memory pool allows a specific number of threads to run.

   With 200 users plus overhead for other threads, such as JDBC™ connections, the recommendation is to start with 300 threads. This is referred to as the *activity level*.

   Allowing the system to adjust the paging characteristics of the storage pool for optimum performance dynamically is also recommended. Setting the paging option to *CALC allows the system to make adjustments as necessary. To change a shared memory pool (*SHRPOOL1) for WebSphere, use the following command:

   ```
   CHGSHRPOOL POOL(*SHRPOOL1) SIZE(x) ACTLVL(y) PAGING(*CALC) TEXT(WebSphere Share
   Pool)
   ```

   In this command, *x* is the total memory allocated, and *y* is the maximum threads based on the user count.

b. Change the subsystem to make use of the shared memory pool.

   The WebSphere subsystem must be able to use SHRPOOL1 with a pool ID of 2. If the subsystem is to run exclusively in the shared pool, run the command:

   ```
   CHGSBSD SBSD(QEJBAS5/QEJBAS5) POOLS((1 *SHRPOOL1))
   ```

   If the subsystem uses memory in the *base pool as well as memory in the new pool, change the subsystem by running the command:

   ```
   CHGSBSD SBSD(QEJBAS5/QEJBAS5) POOLS((2 *SHRPOOL1))
   ```

   Then, you must change each routing entry that needs to point to the new pool by entering the **WRKSBS** command. Take option 5, *Display the subsystem description*, then option 7, *Work with routing entries*. Note the entries that you need to change and enter:

   ```
   CHGRTGE SBSD(QEJBAS5) SEQNBR(nnn) POOLID(2)
   ```

   In this command, *nnn* is the sequence number for each entry that you need to change.

After you make these changes, restart the subsystem. Use the Work with System Status (**WRKSYSSTS)** command to look at the new pool. Validate the shared memory pool settings by using the **WRKSYSSTS** command, pressing the F11 key, and verifying that the “wait to ineligible” value (Wait → Inel) remains at zero.

Note: iSeries Navigator contains a graphical interface that you can use to look at the shared pools.

- Adjust the number of threads running in the JVM

  WebSphere maintains a pool of threads that processes requests from Web clients. Two things limit the pool: a minimum number of threads always started and a maximum number of threads that the pool can grow. You should set the minimum size to handle the average load on the system and set the maximum size to handle the peak load. A good

place to start is to set the minimum number of threads to 10% of the number of users and set the maximum number of threads to equal the actual number of users plus one.

The pool settings limit the number of users that can be handled concurrently. Setting the minimum too high causes additional memory usage. Setting the maximum too low causes users to wait, increasing response times.

Manipulate the thread settings by doing the following:

a. Expand servers and select **Application Servers** → **Your Server Name** → **Web Container** → **Thread Pool**.

b. Change *Minimum Size* to the average workload and *Maximum Size* to the peak workload plus one (necessary overhead for JAS).

c. Select **Is Growable** so that thread allocations beyond the maximum are allowed.

Figure 3-11 shows the recommended values for a maximum workload of 200 users. The minimum size is set to 10% of the total number of users (or 20 threads). The maximum is set to 200+1 (or 201 threads). Also, *Allow thread allocation beyond maximum size* should be selected.

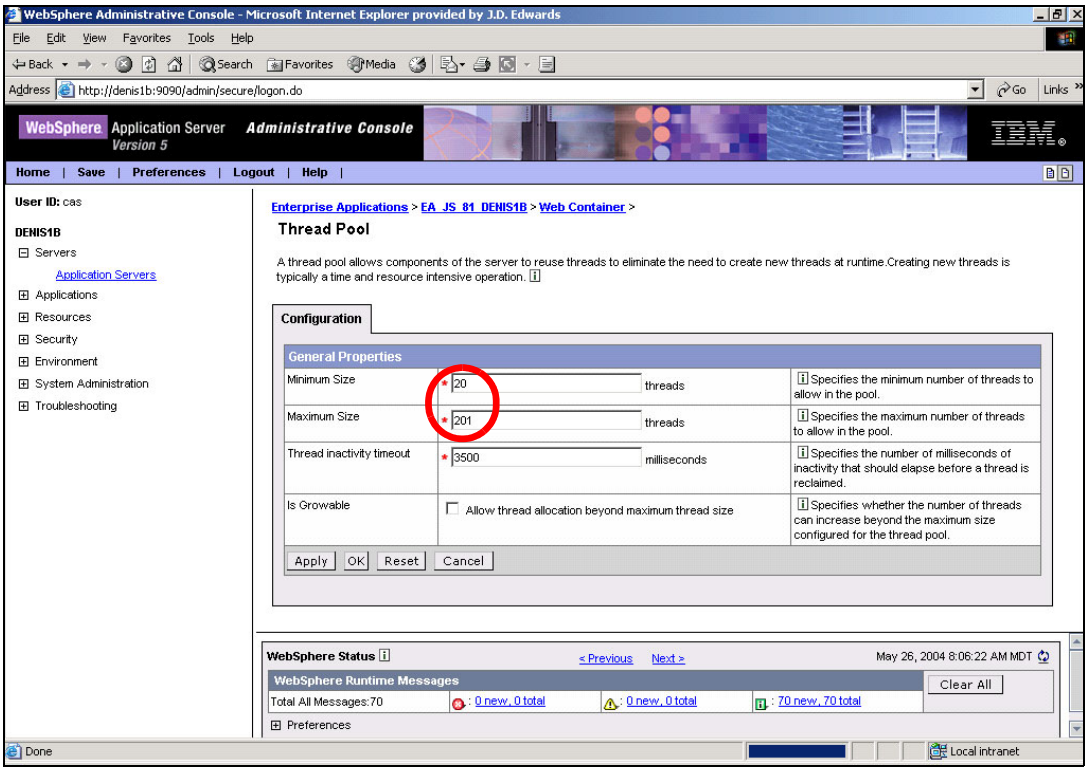

Figure 3-11 Setting the number of threads for WebSphere

Note: Timestamps within the WebSphere and JAS logs are output in Greenwich Mean Time (GMT) by default. To change the timestamps to the current time zone, change the Java system settings. From the OS/400 command line, enter the following command:

```
edtf '/qibm/userdata/java400/systemdefault.properties'
```

Enter the following line to change the time zone:

```
os400.user.timezone=xxx
```

In this command, *xxx* is the correct time zone.

3.3 Tuning EnterpriseOne application performance

JD Edwards EnterpriseOne provides a complete suite of modular, integrated industry-specific business applications. The applications run in a Windows or Web interface. EnterpriseOne is flexible and permits the database, application logic, and user interface code to run entirely on one system or each to run on separate systems. Although some of the tuning recommendations apply to the Windows interface, the information in this section is intended for, and applies specifically to, the EnterpriseOne Web interface.

3.3.1 Measuring and analyzing performance

EnterpriseOne also has built-in tools for monitoring the performance of the user's interaction called the Server Administration Workbench (SAW). Similar to the Apache HTTP server and WebSphere Application Server, the SAW tool uses a Web interface to allow an administrator to monitor the performance of the JAS server and its interactions with the other components of the application and system. SAW provides various views for monitoring:

- System Summary, which shows a high-level view of the heap, including the number of users, business functions, and server uptime.
- User List, which shows the number of users signed on and active.
- CallObject Info, which lists the business functions that have been run and includes statistics about the calls.
- Work with Log files, which contains the logs and .ini files.
- Environment, which lists the various operating system and Java properties.
- Config, which shows some of the jas.ini configuration settings.
- JDENET pool, which lists all of the connections to the enterprise server.
- Work with JDBj, which contains details of the JDBC connections and caches that the JAS server maintains.
- Virtual Clients, which lists all virtual client sessions and activity within them.
- Outstanding Requests, which shows any requests that are currently waiting for the enterprise server to return.
- Thread List, which shows all threads running within JAS and their current activity.

Each of these views is useful for tracking various aspects of the JAS server and its configuration. For example, the System Summary view shows how many users are currently active on the system and how much memory they are using. The other views provide additional relevant information. You can access SAW for JAS servers at:

```
http://<sys>:<port>/jde/saw/sawJAS.html
```

In this URL, <*sys*> is the name of the JAS server, and <*port*> is the port on which the JAS server is listening. Enter the name of the enterprise server for the host. Enter the port on which the JAS server is listening.

The application server portion of EnterpriseOne also contains a SAW function with useful information about the jobs that are running on the enterprise server. You can access this function by running the SAW command from an OS/400 command line. The JD Edwards EnterpriseOne system library, such as B9SYS, must be in the library list for this command to work properly. A Web version of the enterprise SAW function is also available at:

`http://<sys>:<port>/jde/saw/sawEnt.html`

In this URL, <*sys*> is the name of the JAS server, and <*port*> is the port on which the JAS server is listening. Enter the name of the enterprise server for the host, and enter the port of the enterprise server (6011 by default).

SAW for enterprise servers also contains multiple views. The server summary view displays information about each of the server jobs, identifies which jobs have a backlog of requests, and allows tracing to be turned on dynamically.

3.3.2 Adjusting configuration parameters

The JD Edwards EnterpriseOne enterprise server provides configurable parameters through the jde.ini file in the IFS directory. On the iSeries command line, enter the command:

```
edtf '/b9sys/ini/jde.ini'
```

Note: For information about the contents of the jde.ini, see Appendix B, "JD Edwards EnterpriseOne 8.9 test configuration files" on page 125.

Additionally, the EnterpriseOne JAS server provides configurable parameters through the jas.ini file in the IFS directory. On the iSeries command line, enter the following command:

```
edtf '/QIBM/UserData/WebAS5/Base/default/installedApps/<sys>/<ent-app>/webclient.war/
WEB-INF/jas.ini'
```

In this command, <*sys*> is the name of the iSeries server, and <*ent-app*> is the name of the enterprise application.

Note: For information about the contents of the jas.ini file, see Appendix B, "JD Edwards EnterpriseOne 8.9 test configuration files" on page 125.

Configuration parameters that you can tune are:

- Kernel settings

  The EnterpriseOne enterprise server makes use of various jobs to accomplish work for a user. Security kernels authenticate the credentials for Web users. Call object kernels run business functions on behalf of Web users. The JAS server communicates with the enterprise server's network kernels.

  The jde.ini file provides two types of settings related to these jobs. The first setting is the number of jobs to prestart when EnterpriseOne services are started. Additional jobs are started if a request arrives and all jobs are currently active but only up to the second setting, which is the maximum number of jobs allowed to run.

  Call object kernel jobs are capable of handling the requests for multiple users. Because each job caches information and uses system resources, starting too many jobs is not

efficient. However, the presence of too few call object kernels causes users to wait for business functions to run and increases response times.

Lab tests have shown that having 10 to 15 users allocated to each call object kernel provides a good balance of resource use and response time. Probably, 15 users is a good place to start. You can use a tool such as SAW for enterprise servers to provide feedback to fine-tune this setting iteratively.

Starting an optimal number of jobs is true for security and network kernels as well. One security kernel job for every 100 users and one network job for every 20 call object kernel jobs (with a minimum of two jobs) provides the best performance. These settings are changed in the jde.ini file, as follows:

- Network kernel jobs:

```
[JDENET]
maxNetProcesses=x (default is 1)
```

- Security kernel jobs:

```
[JDENET_KERNEL_DEF4]
maxNumberOfProcesses=x (default is 1)
numberOfAutoStartProcess=x (default is 0)
```

- Call object kernel jobs:

```
[JDENET_KERNEL_DEF6]
maxNumberOfProcesses=x (default is 10)
numberOfAutoStartProcess=x (default is 0)
```

- Check kernel health

  While the enterprise server is running, it passes messages between the network jobs and the kernel jobs. By default, every tenth time it sends or receives a message, the network job verifies that each call object kernel is running. Increasing this setting reduces the amount of overhead within the enterprise server and improves overall response time. For example, you can choose to check kernel health for every 5 000 or 10 000 messages. The trade-off is better performance versus prompt knowledge of a kernel problem.

  Monitoring kernel jobs is important, however if you have a heavy transaction load and a large number of call object kernel jobs, you might consider increasing this number to have it check for kernel health less frequently. Alternatively, reduce this number while testing new modifications.

  In the jde.ini file add a new line to the [JDENET] section. Add a line with the setting checkKrnlHealth and enter a value. Setting the network job to check every 5 000 times it sends or receives a message is a reasonable place to start:

```
[JDENET]
checkKrnlHealth=x
```

- Maximum users for the Java server

  The maximum users setting in the JAS acts as an upper limit. Once the maximum number of users has been reached, no additional users are allowed to login to EnterpriseOne until another active user logs out.

  Set the MAXUser value equal to the maximum number of users to be concurrently connected to the JAS. In the [OWWEB] section of the jas.ini file, replace the *x* to set the maximum number of users expected:

```
[OWWEB]
MAXUser=x
```

  Go into Web SAW under the User List menu, verify that your MAXUser setting is high enough by comparing it to the number of users that are connected during peak activity.

- JDENET connections for the Java server

  The connections between the EnterpriseOne Java server and the enterprise server are TCP/IP-based and are referred to as JDENET connections, the same as fat client or TSE users. The number of JDENET connections should be the same as the number of users being run on the Java server.

  To increase the number of connections, add a line to the [JDENET] section of the jas.ini file with a maxPoolSize setting. Replace the *x* with a count of the maximum number of users:

  ```
  [JDENET]
  maxPoolSize=x
  ```

  Go into Web SAW under the JDENet Pool menu to verify the setting. During peak activity, monitor the waiting column. This number should usually be zero (0) to ensure that no latency problems exist for JDENET connections.

- Java server Timeout Setting

  The Tech Flash on Oracle PeopleSoft Customer Connection (GSSTF-02-0392) explains how the UserSession value in the [CACHE] section of the jas.ini needs to be longer than WebSphere's timeout. If not, users can experience errors and have to log out and log back in again.

  You can check the invalidation timeout settings in the WebSphere Administrative Console by following these steps:

  a. Expand servers.

  b. Select **Application Servers** → **Your Server Name** → **Web Container** → **Session Management**.

  c. Note the "Session timeout" setting. The default is 30 minutes.

  Alternatively, you can also:

  a. Select **Applications** → **Enterprise Applications** → **Your Enterprise Application Name** → **Session Management**.

  b. Note the "Session timeout" setting. The default for both the server and the application is 30 minutes.

  The default for the value in the jas.ini file for UserSession is 1 200 000 milliseconds (20 minutes). Change it to 2 400 000 (40 minutes), for example.

  ```
  [CACHE]
  UserSession=x
  ```

  In this command, *x* is the timeout value in milliseconds.

3.4 Tuning iSeries database performance

DB2 Universal Database™ (UDB) for iSeries is an advanced, 64-bit Relational Database Management System (RDBMS). The tight integration of DB2 UDB with OS/400 provides unique attributes to this RDBMS. Single-level storage and the object-based operating system, OS/400, minimize the effort that is required to manage the database, while maintaining its mainframe-like reliability and security.

Automation of many of the common Database Administrator (DBA) tasks that are required by other RDBMS is a cornerstone of DB2 UDB for iSeries lower Total Cost of Ownership (TCO). Because DB2 UDB for iSeries is part of OS/400, many of the tools used to analyze the database are also part of the OS/400 command set.

The EnterpriseOne environment performs the majority of its database access through the following interfaces:

- Open Database Connectivity (ODBC)
- Java Database Connectivity (JDBC)
- Extended Dynamic Remote Support (EDRS)

Each of these interfaces is used by different parts of EnterpriseOne. Table 3-1 shows the different database interfaces, the parts of EnterpriseOne that use those interfaces, and the jobs and subsystems that execute the SQL requests for these interfaces.

Table 3-1 Database connection types

Connection Type	Used by	Job(s)	Subsystem
ODBC	Fat or TSE clients	QZDASOINIT	QUSRWRK
Toolbox JDBC	JAS running on machine other than database server	QZDASOINIT	QUSRWRK
Native JDBC	JAS running on database server	QSQSRVR	QSYSWRK
Local EDRS - First connection per job	EnterpriseOne application server on same system as database (AIO)	EDRSQL interface (no DB job visible in WRKACTJOB)	--
Local EDRS - Additional connections per job	EnterpriseOne job that requires a subsequent connection to database (AIO)	QXDARECVR (one job for each connection) QXDAEDRSQL (one job)	QSYSWRK
Remote EDRS	EnterpriseOne application server on different system than database (V3T)	QXDARECVR (one job for each connection) QXDAEDRSQL (one job)	QSYSWRK

3.4.1 Measuring and analyzing performance

You can use the iSeries Performance Monitor and Performance Explorer utilities to monitor the overall system and database performance. To look more closely at the database component itself, use the database monitor tool. You can start the database monitor through iSeries system commands or through iSeries Navigator. The recommendation is to use the iSeries Navigator interface.

After starting iSeries Navigator:

1. Expand the system.
2. Expand the Databases section, and then expand the local directory (typically this is the system name), as shown in Figure 3-12 on page 46.

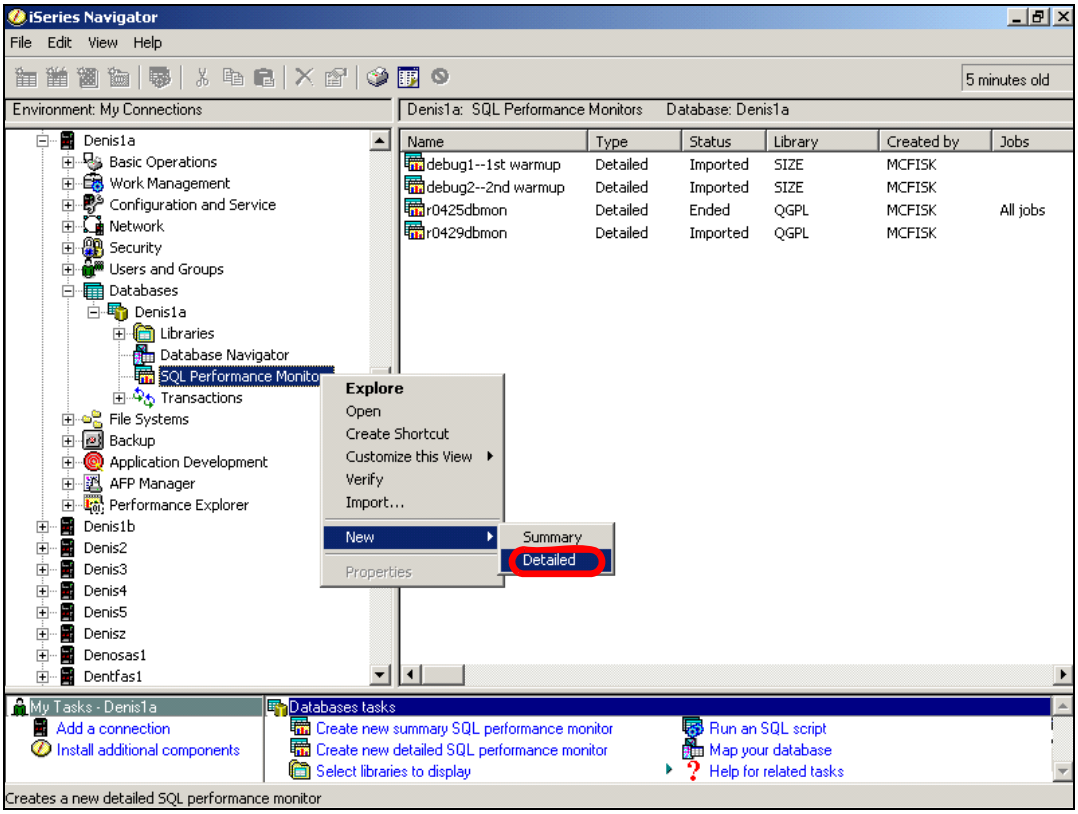

Figure 3-12 Starting the SQL performance monitor through iSeries Navigator

3. Right-click **SQL Performance Monitors**, and select **New**.
4. Select **Detailed** to start the database monitor.
5. On the General tab, give the monitor job a name and a library for storing the data. On the Monitored Jobs tab, select **All** (unless specific database jobs have been identified).
6. Click **OK**. The monitor job is then displayed in the right pane of the window.

   The monitor impacts the user performance while it is running. Therefore, it should be run for a short period of time (10 to 20 minutes).
7. Capture a time period when the system is running the workload of greatest concern, wait for the time period, then right-click the job in the right pane and select **End**.

Alternately, you can use the following commands to start and stop the database monitor:

```
STRDBMON OUTFILE(QGPL/RO429DBMON) JOB(*ALL) TYPE(*DETAIL) COMMENT('April 29 DBMON
information')
ENDDBMON JOB(*ALL)
```

You can import database monitor information that you capture in this way into the iSeries Navigator interface. Right-click **SQL Performance Monitors** and select **import**. Then, enter the name of the monitor and the library that contains the data.

3.4.2 Adjusting DB2 UDB parameter settings

DB2 UDB on iSeries has several settings that you can change to improve the performance of the EnterpriseOne Web client environment. Some of these settings are operating system settings, while others are settings within the Web client configuration files. You can:

- Improve database access performance by creating additional indexes

  Using the information collected by the database monitor as described in 3.4.1, “Measuring and analyzing performance” on page 45, iSeries Navigator can perform various queries on this information. You can use these queries to identify temporary indexes that the system is creating for SQL statements, other SQL statements that require sorting, or long running queries. In addition, iSeries Navigator has a function to explain how specific SQL statements are being executed and another function to create a new index automatically to improve the performance of a query.

  You should use indexes created through the iSeries Navigator interface only temporarily to test the index to see if it improves the performance of the query. Delete indexes that help performance, and then recreate them through the EnterpriseOne Table Design Utility. Using EnterpriseOne to create indexes ensures that they are maintained by the application and are not lost when the application is upgraded.

  Note: You can find additional information about the database monitor in *Using AS/400 Database Monitor and Visual Explain to Identify and Tune SQL Queries*, REDP-0502.

- Tune the number of database connections for the system

  OS/400 maintains prestart jobs for database connections. A prestart job starts before a work request is received, either when the subsystem starts or as a result of the Start Prestart Jobs (**STRPJ**) command. Prestart jobs start from a prestart job entry in the subsystem description. The prestart job entry specifies properties such as what program to run in the prestart job, the memory pool in which the prestart job runs, the number of jobs to start, the threshold at which new jobs are started, the number of times the job is reused, and the number of additional jobs to start.

  The EnterpriseOne Web client environment makes use of database prestart jobs. If a single system is run with the WebSphere application server, the EnterpriseOne application server, and the EnterpriseOne database (All-In-One or AIO), then QSQSRVR prestart jobs are used. If the WebSphere application server is on a separate system (Virtual Three-Tier or V3T), QZDASOINIT prestart jobs are used. All Windows clients use ODBC, which requires QZDASOINIT jobs. Regardless of the type of environment and the type of prestart job, the process of tuning the number of database connections for the system is consistent.

  The system maintains the prestart jobs and starts additional jobs, if necessary. In addition, every five minutes, OS/400 determines whether there is an excessive number of prestart jobs that are not being used. If there is, the OS/400 gradually starts ending the extra jobs until the number is reduced to the number of jobs specified in the initial start setting. Starting and stopping these jobs can cause extra activity as the system attempts to maintain the proper number of jobs. To see the current settings on the database system, use the following command and select option 10 to Display the QUSRWRK subsystem, or enter the following command:

  ```
  DSPSBSD QUSRWRK
  ```

  Option 10 displays the various prestart jobs for the subsystem. Then, use option 5 for the program QZDASOINIT. (For an AIO environment, substitute QSYSWRK for the subsystem name and QSQSRVR for the job name in this example.) Leave the setting (MAXUSE) for the parameter *Maximum number of uses* at 200 (the shipped value). Allowing the

QZDASOINIT jobs to be reused 200 times minimizes the number of jobs starting and stopping (specified with the MAXUSE parameter).

Determine the number of prestart jobs currently being used during the peak periods. Use the Display Active Prestart Jobs command (**DSPACTPJ**) as shown in Figure 3-13 to show the current and peak number of prestart jobs. The command to use is:

```
DSPACTPJ SBS(QUSRWRK) PGM(QZDASOINIT)
```

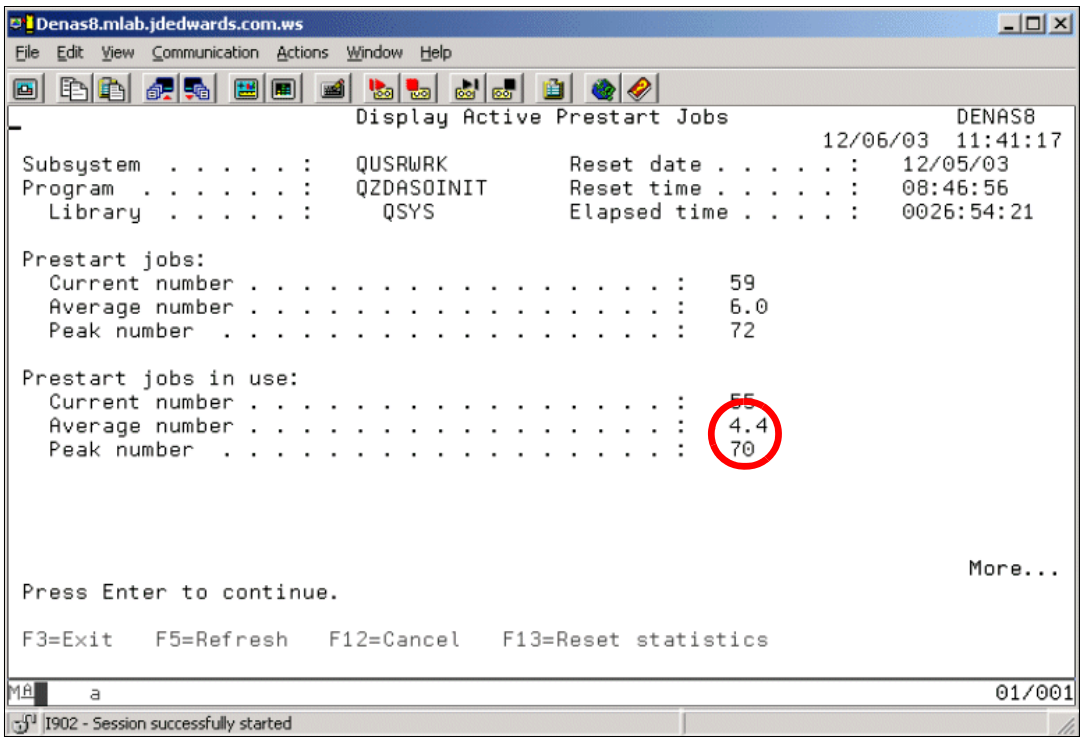

Figure 3-13 Using the DSPACTPJ command to determine the average and peak number of jobs

The key number is the peak number of prestart jobs in use. Use this number to make changes to the prestart job. By setting the initial number of jobs to start so that it is equal to the peak number of jobs, the additional work of starting new jobs is avoided. The jobs are started when the subsystem starts and are available as the connection requests are made. Use the **CHGPJE** command to make the necessary changes for the environment by entering the command and pressing F10 to enter additional parameters, as shown in Figure 3-14 on page 49. Alternatively, enter the complete command:

```
CHGPJE SBSD(QUSRWRK) PGM(QZDASOINIT) INLJOBS(70) THRESHOLD(5) ADLJOBS(10)
```

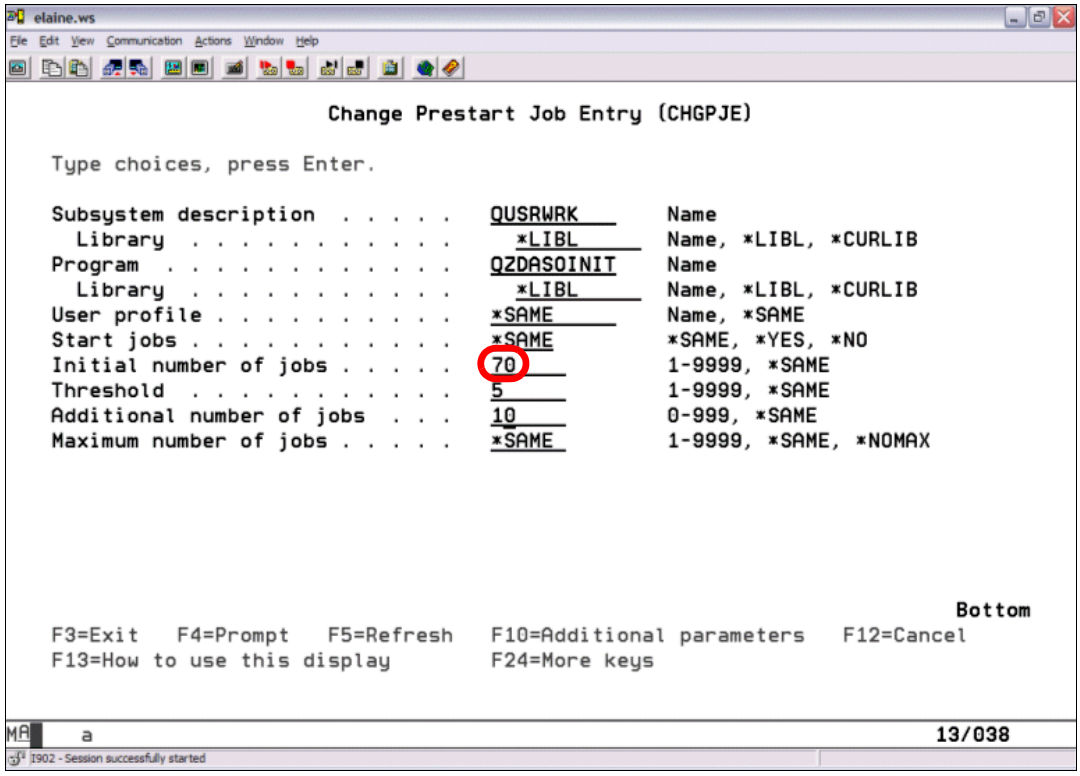

Figure 3-14 Using the CHGPJE command to set new values

For database prestart jobs, the recommended value for Threshold is 5 and for Additional number of jobs is 10. The recommendation for the Initial number of jobs is system-dependent, based on what the DSPACTPJ command shows. In this example, the system prestarts 70 jobs. When 65 jobs are in use, the threshold of five available jobs is reached, and the system automatically prestarts 10 more jobs. If the parameter, Maximum number of uses (MAXUSE) needs to be returned to the shipped value of 200, press F10 for additional parameters and page down to enter the new maximum value.

- Tune the number of database connections for the Web client environment

The EnterpriseOne JAS uses JDBC connections to handle its database work. A JDBC connection is made for a specific iSeries user profile with a given set of properties. EnterpriseOne makes use of proxy profiles that allow many EnterpriseOne users to access the iSeries server through a common user profile. Because these proxy user profiles are used, EnterpriseOne creates pools of JDBC connections that many users can share. A connection pool is started and maintained by JAS for every connection type, where a connection type is the proxy profile and a set of connection properties.

The settings for the JDBC connections are maintained in the JDBj.ini file, which is located in the WEB-INF directory. Typically, this is:

`/QIBM/UserData/WebAS5/Base/Default/InstalledApps/<sys>/<ent-app>/webclient.war/WEB-INF`

In this directory path name, *<sys>* is the name of the iSeries system, and *<ent-app>* is the name of the enterprise application installed in the WebSphere Application Server (or JVM).

To determine the number of JDBC connections required for the pool, look at the Web SAW utility provided with the EnterpriseOne Web client. Pay particular attention to the number of connections being used relative to the initial setting in the JDBj.ini file.

Similar to the prestart jobs, set the initial number of connections to match the number of connections being used. This provides the best performance. Make changes to the [CONNECTION POOL] section of the JDBj.ini file.

```
[CONNECTION POOL]
minConnection=5
maxConnection=x
initialConnection=y
poolGrowth=10
```

In these changes, *x* is equal to the number of Web clients, and *y* is equal to the number of entries (connections) that are displayed in the Web SAW utility column, “Number of entries,” on the Connection Pools Statistics screen. If the maximum connections value is reached, the next user is not denied. However, response times begin to degrade.

Note: For information about the contents of the JDBj.ini file, see Appendix B, “JD Edwards EnterpriseOne 8.9 test configuration files” on page 125.

In our test environment, one connection for every 10 users was a good number for the initial connections setting and a maximum connections setting of one connection for every user was also a good number. A minimum of five connections and pool growth of 10 connections was used in out tests. These settings are a good place to start for tuning an environment. Then, you can make adjustments by observing the Web SAW utility.

A consideration is the use of the Multiple Application Framework (MAF), which allows each user to open multiple windows and potentially run multiple applications simultaneously. It is difficult to predict the impact of MAF on an environment as only a percentage of the users can utilize this function. It is safe to assume that the maximum number of connections should increase with the use of MAF.

If the EnterpriseOne environment is not making use of proxy profiles and each user is mapped to its own iSeries server user profile, the recommendations are considerably different. In this case, set the initial number of connections to one, minimum connections to one, pool growth to one, and maximum connections to 10.

```
[CONNECTION POOL]
minConnection=1
maxConnection=10
initialConnection=1
poolGrowth=1
```

Because of MAF, the maximum number of connections is set to 10, which allows for a single user to have 10 applications open, all utilizing JDBC connections simultaneously.

- Optimize the use of SQL Packages

  SQL packages are OS/400 objects that contain both the control structures and the access plans that are necessary to process SQL statements on the application server when running a distributed program, such as EnterpriseOne. Because SQL packages are shared resources, the information built when a statement is prepared is available to all the users of the package. This saves time, especially in an EnterpriseOne Web client environment where many of the users execute the same or similar queries.

  All the types of database interfaces used by EnterpriseOne make use of SQL packages with the exception of the Native JDBC driver (used with AIO environments). There are two important considerations when using SQL packages in an EnterpriseOne Web client environment:

  – Knowing when to delete SQL packages
  – Ensuring SQL packages are being used effectively

Packages must be deleted when the underlying metadata for statements stored in the package is changed. If a table, view, stored procedure, or other SQL object is altered, information in the package is not updated. Therefore, delete EnterpriseOne SQL packages whenever significant changes have been made to the database, operating system, or hardware. Examples of significant changes are those that can cause a large amount of access plan rebuilds, such as applying a database group PTF. Because SQL packages are recreated when the EnterpriseOne application is run, there is little harm in deleting them.

The JAS code creates a separate SQL package in each library it accesses with JDBC. It is more efficient to use only one SQL package for all JDBC access. This setting is in the JDBj.ini file in the [JDBj-RUNTIME PROPERTIES] section. Add a new setting to force all JDBC access to use a single library for storing SQL packages:

```
[JDBj-RUNTIME PROPERTIES]
as400PackageLibrary=QRECOVERY
```

These settings add the JAS SQL packages to the same library that is already used by the EnterpriseOne kernel jobs. The SQL package JDBJ*xxx* (where *xxx* is a system-generated suffix, typically BBA or BAA) is initially created during the Java-serialized object generation. Once the installation is completed, delete this SQL package, because it contains installation-specific information that is not necessary for normal Web client sessions.

EnterpriseOne kernel jobs use SQL packages created in the QRECOVERY library. Each time the enterprise server is ended and restarted, new SQL packages are built and associated with the new kernel jobs. Old SQL packages associated with kernel jobs that have ended are never used again and can be deleted to save disk space. When enterprise services are down, use the following commands to delete all the SQL packages:

```
DLTSQLPKG SQLPKG(QRECOVERY/O*)
DLTSQLPKG SQLPKG(QRECOVERY/T*)
DLTSQLPKG SQLPKG(QRECOVERY/JDBJ*)
```

Never delete SQL packages that start with the letter “Q.” These SQL packages are used by the system.

Note: OS/400 V5R1 (and higher) supports a call to the job termination signal handler for a job (if one is enabled) when you run the **`ENDJOB OPTION (*IMMED)`** command. If you must use the **`ENDSBS OPTION(*IMMED)`** command, it is recommended that you enable this support, and set the amount of time available for handling the job termination signal to an appropriate value. To set the value, create a data area that specifies the number of seconds for the handler to wait. It is recommended that you initially set this value to 600 seconds if you use the `*IMMED` option to end the QEJBAS5 subsystem or QEJBASND5 subsystem:

```
CRTDTAARA DTAARA(QSYS/QENDJOBLMT) TYPE(*DEC) LEN(5 0) VALUE(600)
```

The maximum value allowed is 3600. The minimum value allowed is 30.

4

Upgrading to JD Edwards EnterpriseOne 8.9 Unicode on iSeries Servers

In today's global economy, many businesses are analyzing the requirements for installing a Unicode database because of its ability to handle multiple languages without resorting to multiple database instances.

This chapter documents the upgrading to JD Edwards EnterpriseOne 8.9 non-Unicode on an IBM @server iSeries server, followed by upgrading the database from non-Unicode to Unicode, as experienced in our test environment. The results include timing the process of the two upgrades as well as tracking the changes in the size of the database. This chapter also includes helpful tips discovered along the way as well as some resource planning guidelines.

© Copyright IBM Corp. 2005. All rights reserved.

4.1 The test environment

The test environment that we used for the writing of this chapter consists of an iSeries Model 810 #2469 2-way processor (2700 CPW) running OS/400 V5R2, with a #2892 PCI Integrated xSeries Server Model 001 (IXS) running Windows 2000 Server with Service Pack 4 (SP4). The iSeries server was installed as an all-in-one configuration, which means that the application, database, Web, and deployment servers were all on the same system, with the IXS card for deployment. Because this was a test environment, only the production environment was installed and measured. There were no custom modifications.

Table 4-1 and Table 4-2 list the hardware and software specifications of the iSeries server where the JD Edwards programs were installed.

Table 4-1 Hardware specifications

iSeries Server hardware specifications	
Model and CPW	810 2700 CPW
Processor	#2469 2-way 750 MHz
Ethernet - iSeries and IXS	1 GB
Memory	16 GB
DASD	15 17 GB drives, for a total of 228 GB

Table 4-2 Software specifications

iSeries Server software specifications		
Operating system release level	OS/400 V5R2	OS/400 V5R2
Software	JD Edwards EnterpriseOne ERP 8.0 SP22_C1	JD Edwards EnterpriseOne 8.9 SP1_B1
Cumulative PTF level	C03161520	C03364520
Hiper PTF SF99519	64	90
Database PTF SF99502	Service Pack 9	Service Pack 11
WebSphere PTF SF99148	Version 4.0.5 Service Pack 6	Version 4.0.5 Service Pack 6
HTTP Server PTF SF99098	Service Pack 10	Service Pack 10

4.2 JD Edwards EnterpriseOne ERP 8.0

The systems were installed with JD Edwards EnterpriseOne ERP 8.0 SP22_C1 by following the JD Edwards installation guide procedures. A sample customer database was used to establish the relative amount of disk storage that was required. Library and directory sizes were collected to set a baseline measurement.

The total disk space that is required for the JD Edwards EnterpriseOne ERP 8.0 installation with one path code is 32.9 GB, as shown in Table 4-3 on page 58. Keep in mind that the test environment was only a subset of the full database load of a normal business environment with multiple path codes.

4.3 The upgrade process

When preparing to upgrade to JD Edwards EnterpriseOne 8.9, review the *JD Edwards EnterpriseOne 8.9 Upgrade PeopleBook for iSeries - Based Systems* manual, also known as the *JD Edwards Upgrade Guide*. You can find this manual, named EO8.9 iSeries Upgrade.pdf, on the CD that is included in the customer software package sent from Oracle.

A quote from the Customer Overview chapter states:

> Unicode specifies that the data stored in the data source is in Unicode format. For installs, all data sources default to Unicode. For upgrades, all data sources but Business Data and Control Tables default to Unicode, which can be converted to Unicode after the upgrade. Unicode is required for Central Objects data sources and, for UDB, any other data sources using the same database as Central Objects.
>
> Note that the code page still must be set to the correct value even though the data sources can all be Unicode. The Unicode flag only indicates what column type is used to store character data in a Unicode database. For example, for DB2 AS/400, it indicates GRAPHIC/VARGRAPHIC with CCSID 13488 is used.

The recommended upgrade process, therefore, is a two-step approach:

1. Upgrade your system to JD Edwards EnterpriseOne 8.9 where all programs and files are Unicode automatically (except for the non-Unicode database, such as PRODDTA and PRODCTL).
2. Convert the PRODDTA and PRODCTL databases to Unicode.

4.4 Upgrading to JD Edwards EnterpriseOne 8.9 (non-Unicode)

The test environment systems were upgraded to JD Edwards EnterpriseOne 8.9 following the instructions included in the *JD Edwards Upgrade Guide*. Installing the deployment server took four hours, as did installing the CDs to upgrade the enterprise server. The JD Edwards EnterpriseOne 8.9 upgrade process, including the Workbench and Table Conversions (TCs), also took about four hours. The client and server package build took less than 10 hours.

For Table Conversion throughput, the default setting in JDEPLAN for the number of threads (queues) performing Table Conversions is four. When initially creating the custom plan, you can change this default. Oracle recommends setting it to one or more per processor on the system, depending on the size of your iSeries server, other workloads, and the network configuration and traffic volumes. In the test lab, since the iSeries server is dedicated to TCs and the dedicated network is one gigabyte, the threads parameter is set to 5 on a 2-way system, which takes the system to 95% processor use.

To modify the threading for TCs, follow the steps in the *Running Table Conversions* section of the *JD Edwards Upgrade Guide*. A summary of the process is:

1. In the Plan, go to the *Advanced Parameters for the Environment* menu and set Table Conversion throttle to *x*, where *x* is the number of threads that you choose for the size of your system. While still logged into JDEPLAN, run **P986130** and set the local queue parameter to *x*. This local queue setting from the Initial Tasks Workbench updates the queue setting on the enterprise server during the initial plan creation.
2. Change the JOBQ QBATCH on the enterprise system to x.

   ```
   CHGJOBQE SBSD(QBATCH) JOBQ(QBATCH) MAXACT(x)
   ```

   To change the queue setting after the Initial Tasks Workbench has been run for the first plan, enter DEP9 and run **P986130** to set the QBATCH default queue with the parameter setting of Max batch jobs =x. Only the first plan modifies the enterprise server.

The Universal Batch Engine (UBE) jobs (500 or so) never appear to be waiting in the JOBQ, so do not look there! To check your progress on the deployment server, drill down, and select the Custom Installation Plan process (GH961 - highlight your plan and select **Expand**). Highlight **Table Conversions** and select **Select**. Use the QBE line and look for conversions where the status is less than 60. Run **P984052** to check for errors in completed TCs. Then search on greater than two in the conversion status.

Also, there is a new SQL Package Library parameter on the host jde.ini file:

```
[DB SYSTEM SETTINGS]
SQL Package Library=1
```

The SQL Package Library parameter value is shipped set to turned on. It needs to be on or active (for example., 1) only during the Table Conversions (TCs) of an upgrade to JD Edwards EnterpriseOne 8.9. The TCs are UBE jobs that all share the same name. If the next UBE uses the previous UBE's SQLPKG, it does not find the framework that it needs, slowing down the upgrade. However, this parameter should be turned off (for example, 2) or commented out for installs, normal operations, and upgrade to Unicode. After the upgrade to JD Edwards EnterpriseOne 8.9 is completed, remember to turn this setting off (for example, 2) in the jde.ini file on the enterprise server.

In releases prior to JD Edwards EnterpriseOne 8.9, there were a few files, including F989998 and F98999 (Central Objects), that need data fields longer than 30 000 characters. The solution was to chain multiple records together to form a single record. The F98BLOB table is used in JD Edwards EnterpriseOne ERP 8.0 and earlier releases to hold this overflow data consisting of the chained records. This solution is informally referred to as the BLOB (binary large object) chaining solution.

With JD Edwards EnterpriseOne 8.9, the F98BLOB table no longer exists, because JD Edwards EnterpriseOne 8.9 and later releases now use a true large object (LOB). Set LOB=True in the jdbj.ini file sections so that the F98BLOB table is not used, as illustrated in Example 4-1.

Example 4-1 The jdbj.ini table

```
[JDBj-BOOTSTRAP SPEC DATA SOURCE]
LOB=True
[JDBj-SPEC DATA SOURCE]
LOB=True
```

During the upgrade, the Table Conversion engine recreates Central Objects files including F989998 and F989999 in a new LOB data source and automatically converts those records using the BLOB chaining solution into true LOBs.

Run the Display File Field Description command (**DSPFFD)** for the F989999 file to verify. If the field WBJPO has a data type of **HEX** with a field length of 30 000 and the notation “Variable length field”, this file is using the BLOB chaining solution. The operating system sets a true LOBs data type to **BLOB**, so if the field **WBJPO** has a data type of “BLOB” and a large field length (larger than 30 000), then this file is using the true LOB solution and the conversion was successful.

Follow the directions in the *JD Edwards Upgrade Guide* in the *Custom Modifications and Packages* chapter to modify the QAQQINI file. Duplicate the file into the QUSRSYS library. Add the following entry to QUSRSYS's QAQQINI file. Without this entry to the QAQQINI file, anything that fetches more than 250 000 LOBs, such as a server package build, fails.

```
LOB_LOCATOR_THRESHOLD 10000
```

Another tip for viewing and editing Integrated File System (IFS) files via iSeries Navigator is to:

1. Select **File Systems** → **Integrated File Systems** → **Root**.
2. Right-click to add a Read/Write Share name of Root.
3. Under Text Conversion, select **Allow Simple Conversion** and add .txt, .log, .ini, .mbr, .sts, and .properties extensions for viewing text files in iSeries Navigator.

As shown in Table 4-3 on page 58, the total disk space that is required for JD Edwards EnterpriseOne 8.9 installation with one path code is 31.9 GB. This is slightly smaller than JD Edwards EnterpriseOne ERP 8.0. In our test environment, the size of the package is smaller, while the database grew 25%.

Table 4-3 Disk space usage

Library and Directory Size (MB)	JD Edwards EnterpriseOne ERP 8.0	JD Edwards EnterpriseOne 8.9 - Non Unicode	JD Edwards EnterpriseOne 8.9 - Unicode
B7334SYS - SP22_C1 / B9SYS - SP1_B1	451	215	215
COPD7334 / COPD9 - Central Objects	2793	3182	3182
DD7334 / DD9 - Data Dictionary	75	240	240
OL7334 / OL9 - Object Librarian	110	170	170
OWINSTALL - OW Save File Library	2837	3425	3425
OWJRNL - OneWorld Journal Library	0	1	1
PD7334 / PD9 - Production Path Code	688	839	899
PD7334DNT - Versions for PD7334	6	0	
PD7334FA / PD9FA - Package Library	3389	1176	1176
PD7334IN / CLARK - Package Library	3271	1757	1757
PRODCTL	120	143	417
PRODDTA	12 121	15 137	20 678
SVM7334 / SV9 - Server Map	6	9	9
SYS7334 / SY9 System Library	97	125	125
Total libraries	25 964	26 419	32 294
/B7334SYS - /B9SYS	55	67	67
/JDEB7334 - JDEB9 logs	1	1	1
/JDEdwards - /PeopleSoft	129	184	184
/OneWorld packages two for ERP 8 two for Release 8.9	4600	3800	3800
/PD7334 - /PD9 specs	2160	1470	1470
Total IFS	6945	5522	5522
Total disk for one path code	32 909	31 941	37 816

4.5 JD Edwards EnterpriseOne 8.9 Unicode conversion

The JD Edwards EnterpriseOne 8.9 customer database is converted to Unicode, following the steps in the JD Edwards Enterprise*One 8.9 Data Conversion PeopleBook*. This section summarizes these steps.

Before converting to Unicode, the QCCSID system value is used to tell iSeries Access to let characters come through without translation. IBM ships the QCCSID system value set to 65535. The JDE user profile's CCSID is 37. Since the test lab installation uses English only, the system value QCCSID is changed to 37 in order that any user profile establishing a database connection can access the data (for example, a user profile that uses the Start SQL command (**STRSQL**) to view the new Unicode sources such as Central Object files, PRODDTA files, and so forth). (The Unicode CCSID is 14388.) It might be preferable to change CCSID at the user profile level, not system-wide.

Note: Environments requiring multiple language support is a different situation and is outside the scope of this Redbook.

To check for Unicode, run a display file field definition command (**DSPFFD**) and look for field types of either CHAR or GRAPHIC. A non-Unicode databases has CHAR field definitions. A Unicode database has Graphic field definitions. For example:

```
DSPFFD PRODCTL/F0002
```

To view logs from within Windows, use iSeries Navigator and navigate as follows:

1. Select **Network** → **Servers** → **TCP/IP** → **NetServer™** → **Advanced** → **Next** → **Start** → **CCSID**.
2. Browse and change the CCSID from `0` to `437`.
3. Stop and restart NetServer.

The steps to convert to Unicode consist of a prerequisite step, three main steps, and a post-requisite step.

The prerequisite step runs an SQL query which lists the obsolete tables in Object Librarian Table F9860. (This cleanup step is designed to save disk space because the subsequent Unicode conversion actually ignores these obsolete tables and any other extra customer files in PRODDTA or PRODCTL). The following is one method to delete these tables from the database prior to converting.

In the test environment, there were 68 obsolete physical files in PRODDTA and none in PRODCTL.

1. Run **WRKOBJPDM PRODDTA** to display the list of all files.
2. Using iSeries Navigator, enter the following SQL command statement to look at the list of 68 files in SQL:

```
select SIOBNM from OL9.F9860 where SISY = '89' and SIFUNO = 'TBLE';
```

Read down the SQL list and delete the logical files followed by the physical file in the green-screen session.

The three main steps after the prerequisite are:

1. Run a local UBE P93091 job on the deployment server to build the Unicode Data Conversion master table from the system and server map data source master tables.
2. Run the data source and table analyzer and conversion program which creates the XML script that defines all the data sources and tables to be converted. This script is the input to the conversion executable:

   C:/B9/script/UniDtaConv.xml

   Evaluate this script for completeness and accuracy because it must run successfully or start completely over again. Ensure that all the data sources to be converted are listed and no extraneous ones have inadvertently been added. The analyzer can automatically add dependent data sources it thinks you need. In the test lab, the XML script included some local databases on the deployment server which caused the script to fail until they were removed.

   Note: Although not recommended, the Conversion Executable step was ended abnormally in the test environment. The chapter entitled *Error Handling* in the *JD Edwards EnterpriseOne 8.9 Data Conversion PeopleBook* was used to assist in the restart. The tables with the suffix "_UNICODE" did not drop as directed. An IPL did not alleviate the problem. It was necessary to run the Reclaim Storage command (**RCLSTG *DBXREF)** to clean up the database cross-reference file. The **RCLSTG** completed in six minutes after going to restricted state.

3. Run the conversion executable after all users have signed off and all applications are down. It converts the metadata from non-Unicode to Unicode and also translates the data from their respective language IDs to a Unicode ID (CCSID).

   ```
   C: B9/system/Bin32/Unicode.exe
   ```

   Check the error log:

   ```
   C:/B9/script/UniDtaConv_Log.xml
   ```

The post-requisite step follows after the conversion executable finishes. At this point, all the physical files are Unicode and there are no logical files. The logical files need to be recreated in PRODDTA and PRODCTL. Start ONEWORLD services and log in to DEP9 on the deployment server to run UBE R9698713. It runs locally to rebuild the logical files. We chose the smaller PRODCTL first, then PRODDTA. When the rebuild of the logical files is finished, follow these steps:

1. Stop ONEWORLD services.
2. Delete all *SQLPKGs (except for QZDAPKG, QSQLPKG2, and QSQXDPKG).
3. Restart ONEWORLD services.

During the Unicode conversion we monitored the temporary space utilized via the Work with System Status command (**WRKSYSSTS**) to determine how much work space the file conversion required. Temporary space grew to the relatively modest size of around three gigabytes. The largest file in the test lab's database is F0911 at 1.5 gigabytes. The conversion used around 44% processor with five QZDASOINIT jobs running.

Unicode conversion times took 3:07 (hours:minutes) to run the conversion script and 1:21 to rebuild the logical files for a total of about 4:30 hours. Although all other libraries and directories remained unchanged (see Table 4-3 on page 58), the database increased 38% over the JD Edwards EnterpriseOne 8.9 non-Unicode database, as shown in Table 4-4 on page 61. The percentage increase in size for a Unicode database is dependent on how much text exists relative to the database as a whole. For example, there are VARCHAR fields which

have variable lengths for text content. If the description fields are very long in general, the percentage uplift for Unicode database sizing can be larger.

Table 4-4 Disk space growth

One path code (MB)	JD Edwards EnterpriseOne ERP 8.0	JD Edwards EnterpriseOne 8.9 non-Unicode	JD Edwards EnterpriseOne 8.9 Unicode	% growth
Total disk usage including database	32 909	31 941	37 816	
Percent growth ERP 8.0 to non-Unicode		-3%		
Percent growth non-Unicode to Unicode			18%	
Total % growth ERP 8 to Unicode				15%
Database only				
PRODCTL	120	143	417	
PRODDTA	12 121	15 137	20 678	
Total database	12 241	15 280	21 095	
Percent database growth ERP 8 to non-Unicode		25%		
Percent database growth JD Edwards EnterpriseOne 8.9 non-Unicode to Unicode			38%	
Total percentage database growth JD Edwards EnterpriseOne ERP 8.0 to JD Edwards EnterpriseOne 8.9 Unicode				72%

4.6 Relative performance

In a separate series of lab tests that were designed to tune and analyze relative performance, standard JD Edwards EnterpriseOne performance tests were executed against JD Edwards EnterpriseOne 8.9 non-Unicode and Unicode databases. The results are compared with previous tests running JD Edwards EnterpriseOne Xe (comparable to JD Edwards EnterpriseOne ERP 8.0). For an understanding of how these standard tests were run, see the document, *BM Benchmarking and Sizing for JD Edwards EnterpriseOne,* posted at the IBM Technical Support page:

http://www.ibm.com/partnerworld/pwhome.nsf/weblook/tech_support_sell.html

Ongoing testing in out test environment continue to yield improvements in performance. For planning purposes, Table 4-5 on page 62 lists the relative resources that might be required when upgrading from JD Edwards EnterpriseOne ERP 8.0 to JD Edwards EnterpriseOne 8.9.

Table 4-5 Processor and memory uplifts

Upgrade planning for JD Edwards EnterpriseOne 8.9		
	Processor percentage uplift	Memory percentage uplift
V3T	22%	10%
JAS	70%	no change
AIO	38%	6%

Virtual 3 Tier (V3T) indicates that an iSeries server with JD Edwards EnterpriseOne Xe or JD Edwards EnterpriseOne ERP 8.0 application and database servers requires 22% more processor and 10% more memory when upgrading to JD Edwards EnterpriseOne 8.9. Java Application Server (JAS) shows that the HTML front-end Web server (a separate standalone iSeries server with WebSphere and HTTP servers) requires 70% more processor and the same amount of memory.

All-in-one (AIO) uplift means that a business whose iSeries server runs application, database, and WebSphere servers all on the same system need to plan on the need for 38% more processor and 6% more memory. The AIO percentages reflect the fact that the enterprise (V3T) workload is weighted slightly more than the JAS when the two workloads are combined.

The performance differences between non-Unicode and Unicode are slight, so Table 4-5 is to document both. Tests run with Unicode database used 4% less processor than non-Unicode.

Note: These guidelines are for planning purposes only. Individual results can vary because of the variety and variability in the installations.

4.7 Summary

In the test environment, the upgrade to JD Edwards EnterpriseOne 8.9 took approximately four hours, an additional four hours to run the Workbench, and just less than 10 hours for a full client and server package build. JD Edwards EnterpriseOne 8.9 required only slightly less disk than JD Edwards EnterpriseOne ERP 8.0 after the upgrade, although the database portion (PRODDTA/PRODCTL libraries) grew 25%.

The data conversion from non-Unicode to Unicode took 4 ½ hours. The Unicode version of the database (PRODDTA/PRODCTL libraries) caused a 38% uplift in disk space that was required. During the conversion, the temporary space growth that was required was equal to approximately double the size of the largest file.

In the test environment, the total database growth (PRODDTA/PRODCTL libraries) when upgrading from JD Edwards EnterpriseOne ERP8.0 to JD Edwards EnterpriseOne 8.9 Unicode was 72%. Because this percentage is dependent on the size of the text in the variable-length fields, results can vary. This information is available as an example of database growth and does not include other factors, such as custom modifications, versions, and PRINTQUEUE.

Although system resources have increased with the introduction of Unicode, the benefits of being able to run multiple code pages and multiple languages, while not having to separate the database into multiple instances are significant.

5

Installing the deployment server on Integrated xSeries Server for iSeries

This chapter describes the unique attributes for the installation process of the Windows operating system when the deployment server is an integrated part of the iSeries server. It is a commentary for the instructions in the iSeries Information Center that you should follow.

The hardware and software required for the deployment server are described in the Minimum Technical Requirements document (MTR) found at:

http://www.peoplesoft.com/corp/en/iou/platforms/enterpriseone.jsp

To access this document, you must first log in to the Oracle PeopleSoft Customer Connection Web site at:

http://www.peoplesoft.com/corp/en/public_index.jsp

Note: The requirements are the same as for an external deployment server.

After verifying the installation requirements for the deployment server, use this chapter to supplement the installation instructions in the IBM iSeries Information Center for the Integrated xSeries Server. You can find step-by-step instructions in the Information Center at:

http://www.ibm.com/eserver/iseries/infocenter

© Copyright IBM Corp. 2005. All rights reserved.

5.1 Overview of the Integrated xSeries Server installation

An integrated server is one of the two following types:

- An Integrated xSeries Server for iSeries (IXS)

  An IXS is essentially a PC on a card. It contains an Intel processor and memory but is diskless. With the IXS, there is a PC card inside the iSeries system that becomes the deployment server.

- An Integrated xSeries Adapter for iSeries (IXA)

  An IXA consists of two components:

  – An HSL bus adapter with cables that are ready to be plugged into a supported xSeries Server

  – One of the supported IBM xSeries Servers, separately ordered, without any disk. This integrated server appears as an HSL attached expansion unit to the iSeries system.

Neither the internal IXS nor the external IXA contain disk drives. OS/400 disk storage is allocated to either server type.

Note: Order the iSeries server with enough disk storage to allow for the disk that is needed to be dedicated to the deployment server.

When an iSeries implementation includes an integrated xSeries server or adapter, the iSeries system arrives with the integrated deployment server in a non-configured state. You can configure the disk as one or more virtual drives, called network storage spaces of varying capacities. These virtual drives are allocated within the OS/400 integrated file system. The Windows software perceives them as physical hard drives. The IXS card is the usual choice for the deployment server because it meets the minimum technical requirements at a lower price than the IXA. The remainder of this chapter, therefore, is based on the IXS.

To calculate the requirements for the size of the deployment server used in Release 8.9, see the *EnterpriseOne Release 8.9 Installation 8.9 JD Edwards* manual on the CD that is distributed with the JD Edwards EnterpriseOne 8.9 software, or log in to the Oracle Web site at:

`http://www.peoplesoft.com/corp/en/public_index.jsp`

The total disk used for the Release 8.9 deployment server is allocated across at least three Network server storage spaces associated with the Integrated xSeries Server. At minimum, the server storage spaces are:

- Windows 2000 server or Windows 2003 server
- Installation source
- Release 8.9 code (based on the number of path codes to be supported)

Note: Windows 2000 is the operating system level described in this chapter. Windows 2003 is also supported.

The first two disk drives (usually named C: and D:), Windows 2000 system and installation source spaces, are created and linked to the network server description during the Install Windows Server (**INSWNTSVR**) command. Consider the following when estimating the size of the C: drive on your Windows server:

- The more memory you have installed on the integrated server hardware, the larger you should make the C: drive. Windows server creates a virtual memory paging file (pagefile.sys) on the system drive. Windows server calculates the size of the paging file based on the amount of physical memory (RAM) installed on the server. It is generally recommended that you allow at least 150% of the installed memory size as additional disk storage on the system drive to accommodate the paging file. For example, on an IXS Server with 512 MB of RAM, you should allow for a paging file size of 768 MB when considering how large to make the system drive.
- When applications are installed on the server, many of them install some of their files on the system (C:) drive, so allow for this requirement.
- When you install a Windows server service pack, you have the option of saving the files that are replaced, in case you need to uninstall the service pack. If you choose to take advantage of the uninstall option, you might require up to 100 MB of additional free space on the system drive.
- Future versions of Windows server might require additional free space on the C: drive to allow for an upgrade of the existing installation.

The following recommendations apply:

- Windows 2000 Server

  C: drive: 1,250 MB + (1.5 x the installed memory) + Application
- Windows 2003 Server

  C: drive: 1,500 MB + (1.5 x the installed memory) + Application

  The D: drive needs to be large enough only to accommodate the Windows Server installation source files, client-side integration software, and any future updates to the integration software (provided by service packs).
- Windows 2000 Server

  D: drive: 500 MB
- Windows 2003 Server

  D: drive: 800 MB

After the **INSWNTSVR** command and Windows 2000 installation is complete, create the Release 8.9 drive and link it to the network server description.

Note: Keep in mind that the total amount of disk that is allocated to network storage spaces is instantly unavailable to the iSeries, so you need to size correctly.

Make the Network Storage spaces large enough to use, but not too large so that there is not enough disk space remaining on the iSeries system for the rest of your installation plans. Track the percent disk utilization for the network server storage spaces by using Work with Disk Status (**WRKDSKSTS**) or iSeries Navigator, both before and after the installation.

For additional details about creating and linking storage spaces, see “Related publications” on page 153.

5.2 Setting up the operating environment

The instructions for setting up the operating environment for this deployment server, prior to installing Release 8.9, are in the IBM iSeries Information Center at:

http://www.ibm.com/eserver/iseries/infocenter

In general, the steps to setup your operating environment are:

1. Choose your region, OS/400 version, and language.
2. Select **Integrated operating environments**.
3. Select **Windows server on iSeries**.
4. Select **Install and Configure iSeries Integration for Windows Server**.

The IBM iSeries Information Center provides the installation, configuration, operational, and general use information for the IXS and IXA features. This topic is also available as a printable PDF file at:

http://publib.boulder.ibm.com/iseries/v5r2/ic2924/index.htm

Note: It is advisable to read all the information that is provided, especially the Pre-installation Checklist. Become familiar with the points concerning the Integration software, hardware, and set up process.

5.3 Installing Windows 2000 or Windows 2003 server

The following steps correspond to those in the topic *Plan for iSeries Integration for Windows Server installation for Windows 2000 or Windows 2003 Server.* These points are the main basis of the installation. Helpful hints are highlighted.

1. Windows Server Installation Adviser

   This step consists of answering configuration questions either via the worksheet or on the Web site using the interactive wizard. After you answer the questions to the Windows server installation advisor, it displays the Planning Worksheet which provides a summary of the values that are specified on the previous panels of the advisor. An example is shown in Table 5-1.

Table 5-1 Configuration questions

Item	Value
Network server description	Name: AS4IXS1 Description: Deployment, Win2k, Lin06, 1.6 GHz, 1 GB memory
Hardware machine type	2892
Resource name	lin06
Windows version	Windows 2000
Language version	Use the primary national language of OS/400
Keyboard layout	Use the default keyboard
Windows source directory	Use the first optical device
System drive (yyyy)	Size: 13 000 MB
Convert system drive to NTFS	Yes

Item	Value
Installation source drive (xxx)	Size: 500 MB
Message logging	Do not log messages
Event log	Log the following: System messages Security messages Application messages
Synchronize time	Synchronize Windows date and time with OS/400 date and time every 30 minutes
Propagate domain users	Propagate domain users using this server
Restricted device resources	Do not restrict tape or optical devices
Client license type	Each computer has a client license
Terminal Services	Do not install
Domain role	Server
Server domain name	Join workgroup: workgroup
TCP/IP local domain name	Use the configured OS/400 local domain name
Private LAN between Windows and OS/400	Use generated IP addresses
TCP/IP	IP address 10.230.14.90
Subnet mask	255.255.255.0
Gateway	10.223.1.124
TCP/IP name server systems	Use the name servers that are configured for OS/400
Clustering	Do not use clustering

Using the wizard is preferred because the end result is the OS/400 command (**INSWNTSVR**) that can be pasted into your green screen.

2. Install iSeries Integration for Windows Server

   This step is unique to the iSeries and a pre-requisite to setting up the integrated deployment server via the installation command. It loads the Integrated Windows software (product number 5722-WSV) and any Program Temporary Fix (PTF) that is required on the iSeries server.

3. Install the Windows server

   This step loads the Windows software using regular Windows CDs similar to an external deployment server, except that the process is begun with the OS/400 command (**INSWNTSVR**) that invokes the Windows installation. The screen that follows summarizes the answers from the advisor onto a command line, which can then be copied and pasted onto your green screen. In our example, the installation summarizes our responses as listed in Table 5-1 on page 66 are summarized in Figure 5-1 on page 68.

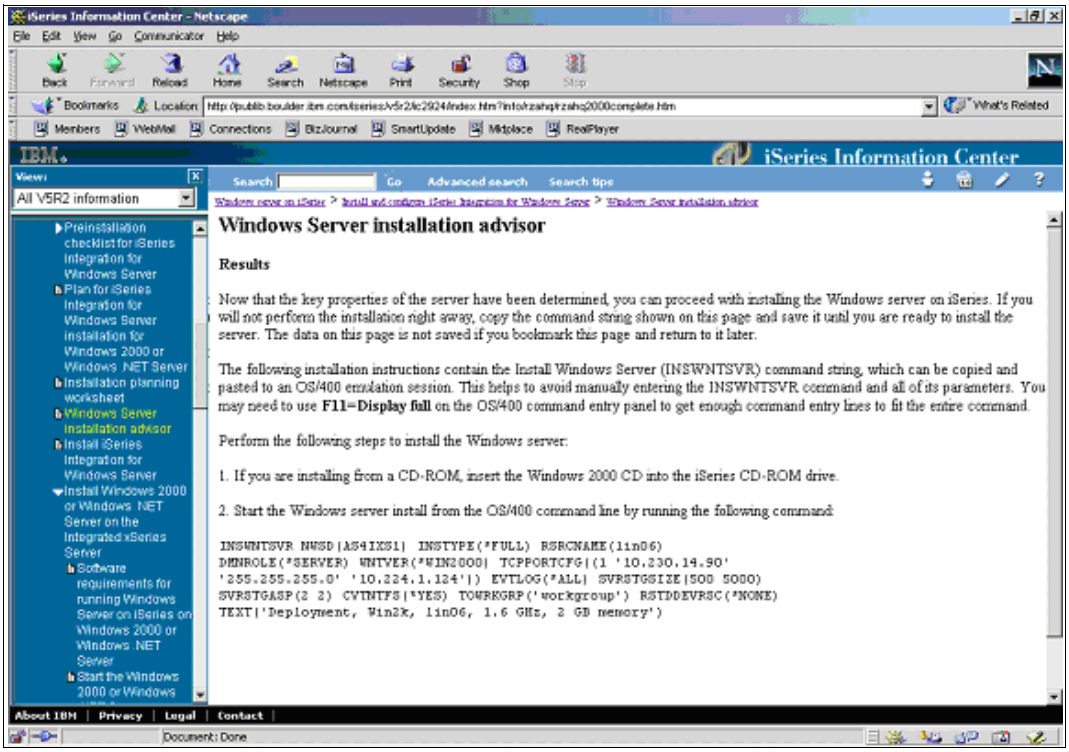

Figure 5-1 Windows Server Installation Advisor results

Note: Before you press Enter for the **INSWNTSVR** command, be sure to have your keyboard, mouse, and work station attached to the IXS card. The command continues by booting the IXS card and instructs you to continue at the Windows server's console. If the hardware is not attached in advance, vary off the IXS, and repeat this step.

Copy the resulting command containing your choices onto your green screen, similar to this one:

```
INSWNTSVR NWSD(your server name) INSTYPE(*FULL) RSRCNAME(your communication line)
DMNROLE(*SERVER) WNTVER(*WIN2000) TCPPORTCFG((1 '10.230.14.90' '255.255.255.0'
'10.224.1.124')) EVTLOG(*ALL) SVRSTGSIZE(xxx yyyy) CVTNTFS(*YES) TOWRKGRP('workgroup')
RSTDDEVRSC(*NONE) TEXT('Deployment, Win2k, lin06, 1.6 GHz, 1 GB memory'
```

In this example, *xxx* is the size of your installation source drive, and *yyyy* is the size of your Windows C: drive. Be sure to put the first Windows install CD into your iSeries CDROM drive, and press Enter. This command loads Windows and creates your C and D drives out of iSeries disk, a C drive for boot and Windows system drive, and a D drive for Windows installation source drive.

Note: The green screen remains input inhibited until the completion indicator is shown and responded to at the Windows server console.

4. Continue the Windows 2000 or Windows 2003 Server Installation from the Windows server console. You must select **Finish** to restart the integrated Windows server.

   Note: Rebooting the IXS card does not affect the rest of the iSeries system.

   From the Windows screen, use normal Windows commands such as the Start Menu's Shutdown or Control+Alt+Delete, restart the IXS card. If the IXS card happens to be powered off at any time, then the IXS card needs to be varied on under Work with Configuration Status (**WRKCFGSTS**) *NWS or under iSeries Navigator's Network, select **Windows Administration** → **Integrated xSeries Servers**, and right click the name of your IXS server.

5. Complete the Windows 2000 or Windows 2003 Server installation. Refer to the JD Edwards documentation on the CD shipped with the Oracle product, or log in to the Oracle web site at:

   http://www.peoplesoft.com/corp/en/public_index.jsp

   Install the recommended Windows Service Pack level from Microsoft that is normally required via Windows.

   An IBM maintenance wizard is used to install the Windows Integration software service pack of PTFs from IBM that gets downloaded to Windows from the iSeries server. This step also includes directions for setting your Windows server to automatically vary on with TCP/IP (including at IPL time).

   The next step is to vary your deployment server on and validate it.

   Note: In this integrated environment, OS/400 and Windows typically share the iSeries CDROM or DVD drive. In order for Windows to use the drive, it must first be locked.

   To lock the CDROM or DVD drive, follow these steps:

   a. Select the Windows Start option.
   b. Select **Programs** → **IBM iSeries** → **Integration for Windows Server**.
   c. Name your IXS server at the prompt.
   d. Select the iSeries devices, and select **OPT01** (or your optical drive name).
   e. Select **All Tasks**.
   f. Right click and lock device.

   Be sure to unlock the device after its use if the iSeries needs it next. Rebooting the deployment server always unlocks the optical device automatically, so be sure to relock it if deployment server will use it again after a reboot.

5.3.1 Adding Network server storage spaces for JD Edwards implementation

After completing these steps, carve out at least one additional Network server storage space from iSeries disk to be used as a dedicated deployment server disk for Release 8.9 with an assigned drive letter. In the IBM Information Center, within the *Windows server on iSeries* topic, go to the *Manage storage* topic, and then Add disk drives for Windows server on iSeries. Carve out the additional network storage spaces for the amount of disk that you have chosen to use at this time.

For the one path code in the test environment, we ran:

```
CRTNWSSTG NWSSTG(AS4IXS14) NWSSIZE(40000) TEXT('Release 8.9 G drive')
```

To link the drive to the Windows server after the storage spaces are created, vary the IXS off again. After the Release 8.9 drive is linked, vary the deployment server back on. Then, proceed with the normal set up of C++, TCP/IP, FTP, and so on.

5.3.2 Operational tips

The following are operational tips after the installation is complete:

- Backup your C: drive into a save file.

  Follow these steps to create a save file (*SAVF) and back up your C drive:

  a. Run the following command:

     ```
     CRTSAVF FILE(QGPL/IXS1CDRIVE)
     ```

  b. Vary off the deployment server.

  c. Run the following command:

     ```
     SAV DEV('/QSYS.LIB/QGPL.LIB/IXS1CDRIVE.FILE') OBJ('/QFPNWSSTG/AS4IXS1') DTACPR(*YES)
     ```

  This backup is used to quickly recover the Windows operating system.

- Increase the size of your application event file on the deployment server (for example, from 512 KB to 2 MB) before beginning the deployment process. This helps prevent it from filling up and halting the job.
- Do not run the **defrag** command. Objects, and parts of objects, are created in non-contiguous sectors, as part of the architecture of the iSeries and OS/400. This eliminates the need to reorganize disk space to consolidate available space, as is required on other platforms.

6

Virtualization for server and application consolidation

As the cost of managing systems continues to increase, consolidation of systems and applications has become an important concern for many businesses. Successful consolidation requires that a server that can handle multiple operating systems and application environments simultaneously and affordably. The IBM @server i5 and iSeries servers help consolidate systems and applications.

The IBM @server iSeries family of servers allow logical partitions to run multiple environments of the IBM Operating System/400® (OS/400) and the 64-bit Linux® on POWER™ operating system. Now, with the availability of the POWER5™ technology-based IBM @server i5 servers, successful consolidation is a more viable option. You can consolidate IBM AIX 5L™ operating systems and 32-bit Linux operating systems (Linux on Intel via Integrated xSeries Adapters or Integrated xSeries Servers) on the same IBM @server i5 server. In addition, Integrated xSeries Adapters and Integrated xSeries Servers support multiple Microsoft Windows operating system based applications.

This chapter describes how IBM uses new features of the IBM @server i5 server to maximize the performance of four applications running under different operating systems. This chapter describes the benchmark and test results in more detail.

Note: This chapter is based on the latest servers and operating system level announced in May 2004 for iSeries customers.

The Virtualization benchmark requires IBM @server i5 POWER5 hardware and, therefore, V5R3 i5/OS. Some JD Edwards EnterpriseOne applications, such as Learning Management, require IBM @server i5 (POWER5) servers with AIX and i5/OS V5R3.

The test environment is further described in Appendix A, “The JD Edwards EnterpriseOne 8.9 test environment and transaction details” on page 119.

© Copyright IBM Corp. 2005. All rights reserved.

6.1 Virtualization benchmark overview

To demonstrate the versatility of new server and software products, IBM has a new benchmark that uses popular commercial applications such as JD Edwards Enterprise One, JD Edwards Enterprise Learning Management, Trade3, Linux file serving (Samba), and a Linux firewall. Known as the Virtualization Grand Slam, the results of this benchmark show how the IBM @server i5 server can deliver outstanding performance with excellent response times.

The benchmark testing pushed the number of users per processor to new levels by taking advantage of POWER5 simultaneous multi-threading capabilities. In addition, the benchmark results show how the IBM @server i5 server can share processors dynamically across logical partitions, allowing the server to handle a sudden increase in application workload in one partition by harnessing under used and available resources in another partition without operator intervention.

The steps of the Virtualization Grand Slam benchmark are:

1. Test each application individually to get a performance baseline.
2. Run all applications concurrently in fixed partitions to show that we can combine these workloads without degrading performance.
3. Show that by sharing processor resources, we can improve overall performance by dynamically allocating resources across partitions. By artificially introducing sudden increases in workload, we demonstrate how under used resources can be used to smooth out the spikes and improve overall performance.

During the course of this benchmark, we recognized that systems and application management is a key aspect of consolidation. New IBM Virtualization Engine™ console software presents a common monitoring interface for heterogeneous environments, to consolidate multiple operating systems and streamline operations. In addition, alerts can be set at various thresholds to inform system administrators or users when a problem occurs.

In summary, the results of the Virtualization Grand Slam benchmark demonstrate how the IBM @server i5 server can provide excellent performance and manageability and help create new opportunities for server and application consolidation.

6.2 Virtualization benchmark goals

The new IBM @server i5 servers help create several opportunities for IBM clients. These include greater processor performance and the ability to run AIX 5L and Linux operating systems in separate micro-partitions side-by-side with the IBM i5/OS, allowing a broader range of applications to run on IBM @server i5 servers.

Systems management has also been simplified through virtualization. The Virtualization Grand Slam benchmark explores hardware virtualization using POWER Hypervisor firmware and new management utilities such as the Virtualization Engine Console system technology.

The Virtualization Grand Slam benchmark test achieves the following goals:

- Demonstrates the capabilities of the IBM @server i5 server as a consolidation platform for multiple applications and operating systems
- Shows how hardware and software applications can be virtualized to make a single server appear as though it were multiple servers

- Demonstrates management of heterogeneous operating systems and application environments
- Uses POWER5 processors to demonstrate the scalability of popular commercial applications

Before beginning this benchmark test, IBM looked at the real-world issues that its clients encounter. The IBM benchmark team uses popular commercial applications and application workloads and runs them in micro-partitions in much the same way a client would. The team also looks at other applications, such as a firewall and file serving, to complete the test suite.

Table 6-1 shows the applications and platforms that were chosen for this benchmark.

Table 6-1 Application workload overview

Application name	Description	Platform	Planned workload	Workload metric
JD Edwards Enterprise One 8.9	Manufacturing, distribution, and financials	i5/OS, IBM DB2 Universal Database (UDB) V5.3	500 users	Processor use and response time
JD Edwards Enterprise Learning Management 8.81	Human resources skills management	AIX 5L V5.3, IBM DB2 V8	600 users	Processor use and response time
Trade3	Stock-trading benchmark	i5/OS, DB2 UDB V5.3	1000 users	Processor use and transactions per second
Samba	File serving	SUSE LINUX Enterprise Server 9 (Linux 2.6 kernel)	25 users	Megabytes transferred per second

In addition to the applications in Table 6-1, all of the Trade3 and Samba data are routed through specific firewall ports to test the Linux firewall. The IBM benchmark team measured Trade3 and Samba throughput both with and without the firewall to investigate the overhead of using a Linux firewall.

For the PeopleSoft and Trade3 workloads, Mercury LoadRunner scripts are used to simulate HTML browser clients.

The two PeopleSoft workloads run inside the firewall, and these workloads are run over Virtual Ethernet technology. The LoadRunner scripts are executed from IBM Integrated xSeries Servers adapters that physically reside inside the IBM @server i5 disk towers, share IBM @server i5 disk storage and use the Microsoft Windows Server 2000 as the operating system.

The Trade3 workload is driven by two large IBM @server xSeries servers that reside outside the firewall and are connected using a 1Gbps optical fiber local area network (LAN). The Trade3 communications traffic is sent to the Linux partition through the Linux firewall and then routed via Virtual Ethernet to the Trade3 logical partition.

The Samba file serving is driven by an xSeries server running as a Linux client and connected to the IBM @server i5 system using the same 1Gbps optical fiber LAN.

LoadRunner scripts are used to simulate HTML browser clients for the JD Edwards Enterprise Learning Management workload. On an additional PC, Mercury QuickTest Professional (QTP) is used to run 13 actual transactions with 10 iterations per transaction, including keystrokes and data entry, to give more accurate response time figures.

6.3 Application and server workload description

There are four independent workloads running concurrently on a single IBM @server i5 server. Each workload runs in one of four separate partitions.

This section describes the following workloads:

- JD Edwards Enterprise One 8.9
- JD Edwards Enterprise Learning Management 8.81
- Trade3
- Samba Server Message Block

6.3.1 JD Edwards Enterprise One 8.9 application

JD Edwards Enterprise One 8.9 application software is used by many diverse companies and is popular in the manufacturing, distribution, and financial markets. The application workload was derived from each of these areas, with 18 transactions based on the typical mix used at real client locations.

The application workload is divided into three user groups: manufacturing, distribution, and financial. In the companies surveyed, approximately half of the users are in the distribution group while the remaining users are split between the manufacturing and financial groups. You can find detailed transaction information in Appendix A, “The JD Edwards EnterpriseOne 8.9 test environment and transaction details” on page 119.

Note: Although IBM used the Unicode version of the database for this workload, a non-Unicode database can be used.

6.3.2 JD Edwards Enterprise Learning Management 8.81 application

The JD Edwards Enterprise Learning Management 8.81 application allows companies to manage skills acquisition. This workload runs 13 transactions, details of which can be found in Appendix A, “The JD Edwards EnterpriseOne 8.9 test environment and transaction details” on page 119. The benchmark measures the number of transactions submitted per user during the elapsed test time.

This workload uses LoadRunner scripts to simulate HTML browser users. To obtain the true average response time, another set of QTP scripts is used on a separate PC to allow the overhead of the PC to be included in the response time. These scripts are run concurrently with the LoadRunner scripts so that response time is calculated while the system is under load.

6.3.3 Trade3 application

Trade3 is an Internet-based benchmark application that simulates an online stock-trading application. The Trade3 application allows a user to:

- Register to create a user profile, user ID, password and initial account balance
- Log in to validate a previously registered user
- Browse current ticker symbols for a stock price
- Purchase shares
- Sell shares from holdings
- Browse a portfolio
- Log out to terminate the user’s active interval

Transactions use Secure Sockets Layer (SSL) to protect privacy.

The Trade3 workload is driven by LoadRunner scripts that emulate HTML browser users. To simulate a real-world deployment, the transactions are routed through a Linux firewall. Response time i derived from response time data collected by LoadRunner.

6.3.4 Samba file sever

The Samba Server Message Block (SMB) file server is a workload based on the NetBench file server workload. NetBench requires several PCs and is expensive to configure and run. The Samba file server workload (sometimes referred to as SMB torture testing) offers an affordable alternative by using captured NetBench primitives, allowing the workload to be driven by a few client PCs or even one PC, as in the Virtualization Grand Slam benchmark.

In this test, throughput is measured in megabytes per second (MB/sec) for a given number of users. The IBM benchmark team routes file server traffic through the Linux firewall to increase the firewall workload.

6.3.5 Virtualization benchmark methodology

Comprehensively testing the IBM Advanced Micro-Partitioning™ (LPAR) capabilities of the IBM @server i5 system required the benchmark team to :

1. Establish individual baselines for each application workload running individually on the system in its own 2-way, fixed partition and scale each workload to the required resource use.
2. Run all application workloads concurrently, each in a 2-way, fixed partition, and compare the results to the baselines obtained in the previous step.
3. Change the partition definition to use shared processors and measure new baselines with all workloads running concurrently, comparing the results to those of the previous step. Some partitions were uncapped, allowing them to use under used resources from other partitions.
4. Significantly increase one of the workloads while all workloads are running concurrently using shared processors to demonstrate resource sharing, and then compare the results to those of steps 2 and 3. This workload was uncapped, allowing under used resources to be “borrowed” from other partitions.

6.4 System architecture

This section describes the architecture of the system.

6.4.1 Virtualization system configuration

The IBM @server i5 570 that is used in this benchmark is an 8-way server with 128 GB of main memory and 237 35 GB disk drives.

The Virtualization Grand Slam benchmark does not use all of the disks, memory or attached hardware. The system is initially configured into four logical partitions, as shown in Figure 6-1.

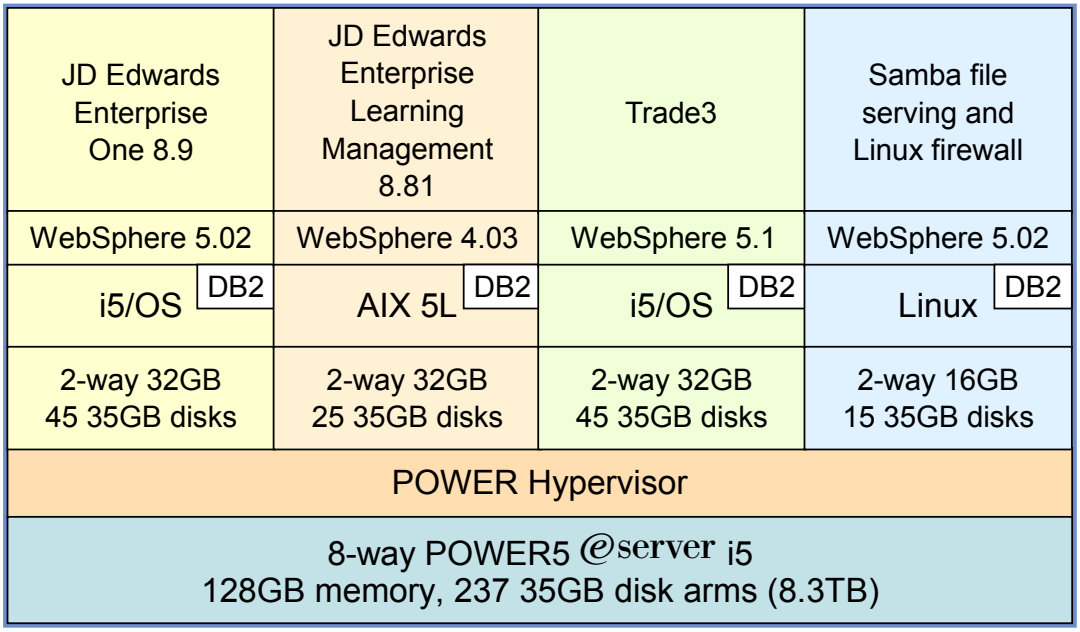

Figure 6-1 Logical partitioning schematic

The Samba workload is run via the Linux firewall to increase load on the firewall. In general, the file server would be behind the firewall and possibly in its own partition.

Each partition uses direct-attach disk drives. AIX and Linux disks are formatted to use RAID-5, giving customers maximum flexibility in setting up their partitions.

IBM POWER Hypervisor firmware virtualizes a physical system so that each application appears to run on its own logical system. In the benchmark test, all disks are connected in RAID-5 sets of either 10 or 15 disks. Note that 16GB of memory and 107 disks are not used in this configuration.

6.4.2 Network configuration

Connections to external and internal LANs were also virtualized in this benchmark test. The applications have varying networking needs depending on their individual natures. Figure 6-2 shows a schematic of the network.

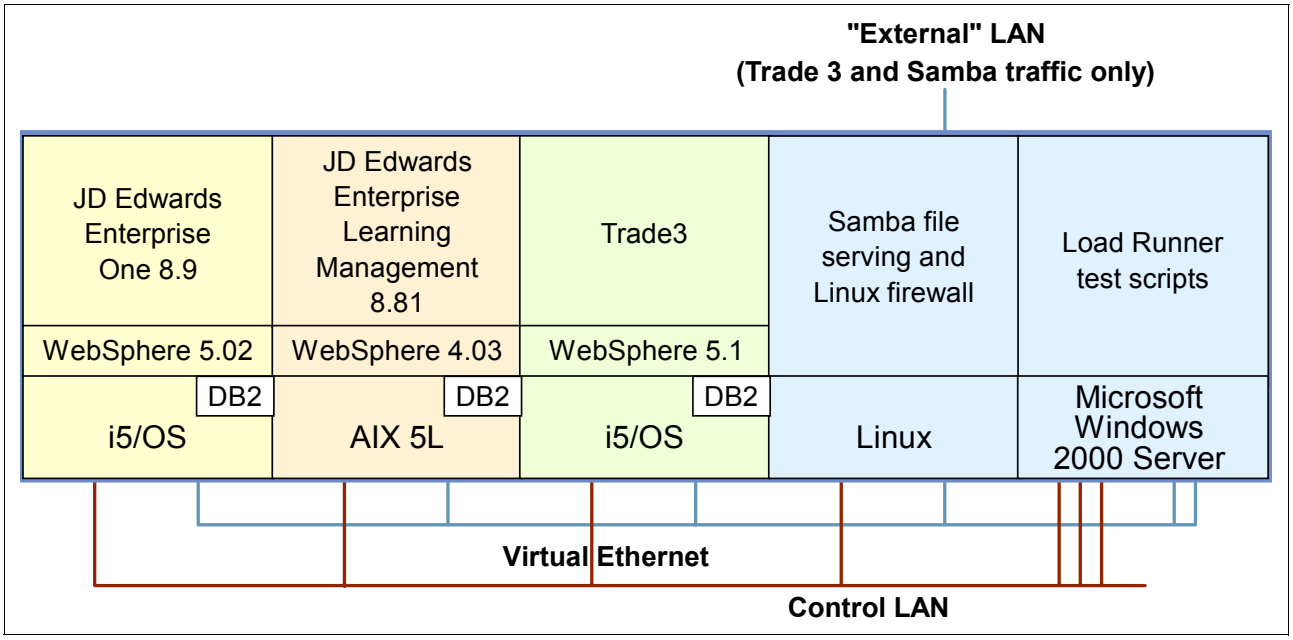

Figure 6-2 Network architecture

Each partition is connected to a Virtual Ethernet that appears to the partitions as a 1Gbps Ethernet adapter. The Virtual Ethernet efficiently uses the system bus to connect partitions, achieving a maximum throughput of approximately 450 MB/second. The JD Edwards Enterprise One and JD Edwards Enterprise Learning Management applications are connected directly to their respective load generators via the Virtual Ethernet. This simulates real-world client transaction systems inside a firewall.

All micro-partitions and Integrated xSeries Servers adapters are connected to a control network for administration, configuration, monitoring, and control.

The Trade3 application and the Samba file server are both protected by a Linux firewall. The load generators for these two workloads are connected via a 1Gbps optical fiber network to the Linux firewall. Specific firewall ports are opened for this traffic only. The Samba file server is located on the Linux partition inside the firewall, while the Trade3 traffic is routed by the Linux partition to the Trade3 partition via the Virtual Ethernet.

In this network configuration, all LANs are lightly loaded (using approximately 5% to 10% of the capacity) because of the high bandwidth. The protocol used is TCP/IP.

6.4.3 Virtualization Engine Console

Systems management across multiple systems is always a challenge. This is compounded when the systems have different operating systems. Virtualization Engine Console software is an IBM Licensed Program Product (LPP) that allows the monitoring of heterogeneous environments through a common user interface. This product plays a key role in monitoring the four logical partitions under test and generates performance-based alerts when exceptions occur.

The ability to collect relevant base data is necessary to display the status and performance of multiple systems. Two LPPs were employed to do this:

- Management Central, which is part of IBM @server iSeries Access for Windows
- IBM Director Multiplatform Agent, which is part of IBM Virtualization Engine systems technologies

In traditional IBM @server i5 and iSeries client sites, Management Central is often used to manage single and multiple IBM @server i5 systems or i5/OS micro-partitions. IBM Director helps clients who run Linux, AIX 5L or Windows manage their different environments.

Figure 6-3 shows how you can use Management Central and IBM Director Multiplatform Agent to collect data.

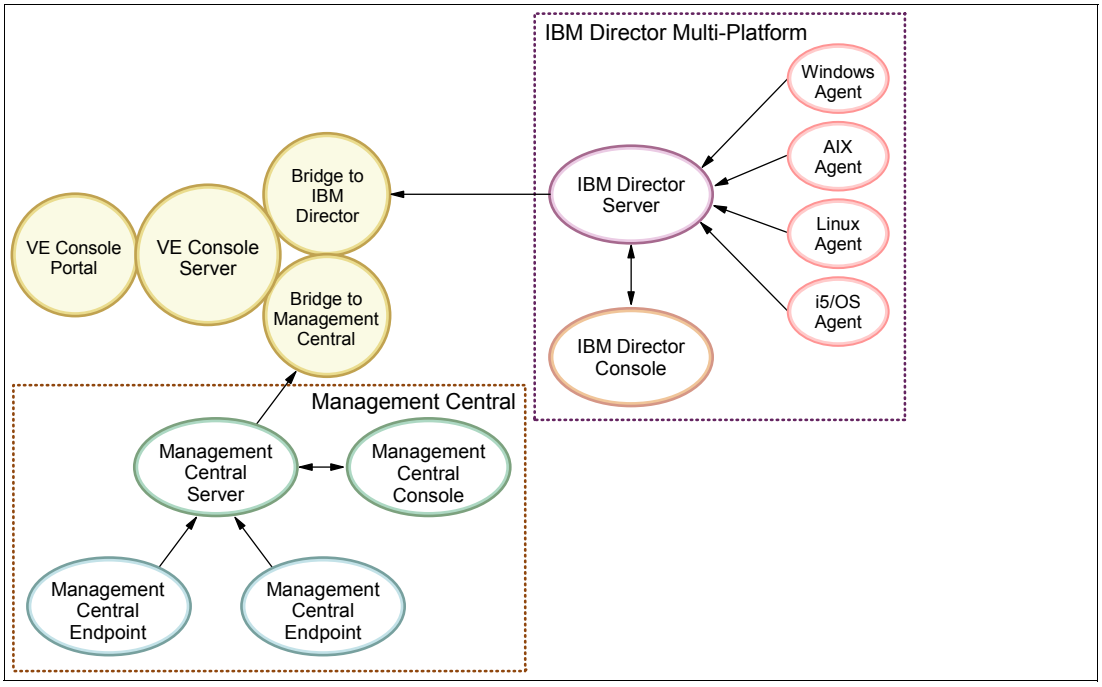

Figure 6-3 Schematic of Virtulization Engine Console implementation

i5/OS systems can use either Management Central or IBM Director Multiplatform Agent to collect data. However, Management Central offers much more functionality than just data collection.

Data is consolidated and stored either in the Management Central server or the IBM Director server. Virtualization Engine Console software can access the data from either type of server using IBM WebSphere technology.

IBM Virtualization Engine Console software uses a standardized WebSphere portal to present information to the user. Figure 6-4 shows a sample screen capture from the Virtualization Grand Slam benchmark test.

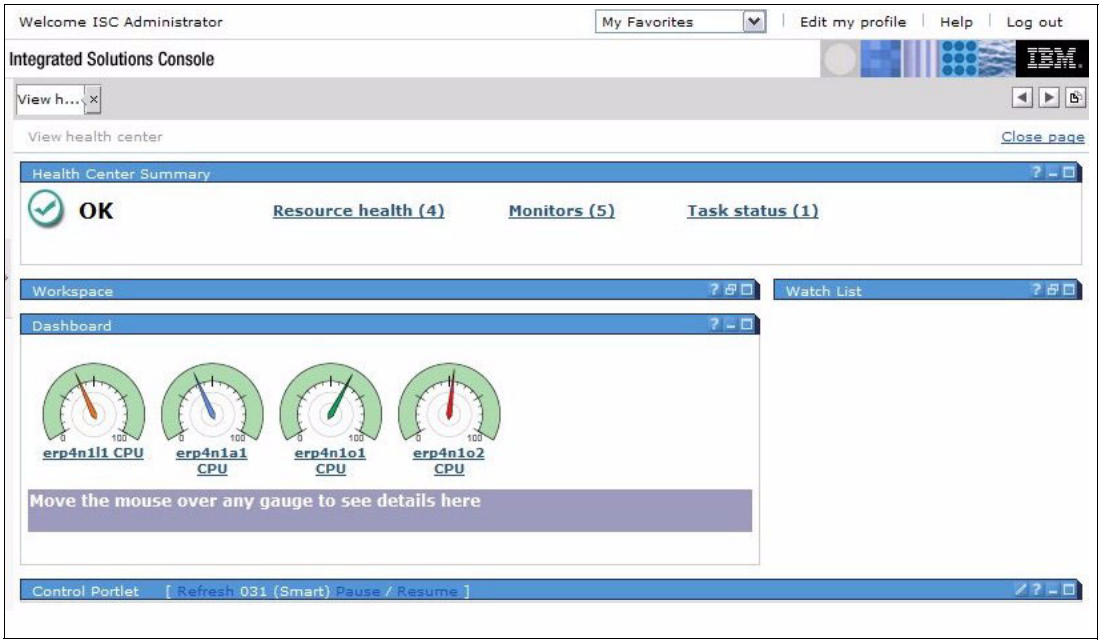

Figure 6-4 Virtualization Engine console sample screen capture

In this example, processor utilization is being monitored for the Linux, AIX 5L, and i5/OS partitions. By using portal technology, you can tailor this interface to suit individual needs. Alerts can also be used to provide warning when thresholds are exceeded.

6.4.4 Virtualization benchmark performance results

Performance can be categorized in several ways, such as by throughput, response time, transactions per second, or processor utilization. In the Virtualization Grand Slam benchmark, there are different metrics for each workload on fixed partitions and overall metrics for the system's total performance, such as processor utilization when using shared partitions to dynamically transfer resources across partitions to handle spikes in workloads.

Fixed partitions

The system is initially split into fixed partitions, each with two processors. The workloads are run standalone (only one workload is run in a partition with no other workloads running), and

then all workloads are run concurrently in fixed partitions, followed by workloads that are run in shared partitions. Table 6-2 summarizes some of the base test results.

Table 6-2 Summary of performance data for fixed partitions

	JD Edwards Enterprise One 500 users	**JD Edwards Enterprise Learning Management 600 users**	**Trade3 1000 users**	**Samba File Serving 25 users**
Standalone 2-way, fixed partition	Response time: 0.7 seconds Processor: 69%	Response time: 3.5 seconds Processor: 63%	Transactions/sec: 66 Processor: 57%	Throughput: 41 MB/sec Processor: 40%
Concurrent 2-way, fixed partition	Response time: 0.7 second Processor: 71%	Response time: 3.6 seconds Processor: 64%	Transactions/sec: 66 Processor: 59%	Throughput: 40 MB/sec Processor: 41%
Concurrent 2-way, shared partition	Response time: 0.4 seconds Processor: 73%	Response time: 3.7 seconds Processor: 70%	Transactions/sec: 66 Processor: 65%	Throughput: 39 MB/sec Processor: 43%

Performance differs minimally when comparing the standalone workloads to the concurrent workloads. The small increase in processor usage is possibly due to minor level 3 cache contentions between processors.

Shared partitions

In shared partitions, all processors belong to a pool. Using the IBM Hardware Management Console (HMC) to configure these partitions, each workload is initially given two processing units (equivalent to two physical processors) and four virtual processors. As changes in workloads occurred, virtual processors are allocated to partitions that need extra processing power from partitions with excess resources.

A nominal increase in processor overhead occurs when moving to shared partitions because the server lost some processor and memory affinity. The increase is workload dependent, ranging from 2% to 6% but averaging about 4%.

The average response time in the most heavily used partition drops from 0.7 to 0.4 seconds when shared partitions are used. Other partition metrics remain similar to those measured using fixed partitions.

To investigate the benefits of shared partitions, the IBM benchmark team introduced a new category of workload that uses a sudden increase in the workload in one partition to measure how quickly that partition adapts to the increase. The team also monitors the effect on the other partitions. When creating the spike, the team expects that spare resources from other partitions are dynamically available to the requesting partition.

One method of creating a spike is to submit a resource-intensive batch job that pushes the system to a processor use of 100% for a short period of time. Depending on the use before the spike, the batch job might run for a longer period of time if the combined resources would normally exceed 100%. By measuring the elapsed time of the batch job and by using the metrics described earlier, the benchmark team is able to determine the ability of the system to balance resources across shared partitions.

Table 6-3 shows the effects of a spike on both a single batch job and the other partition's metrics.

Table 6-3 Summary of performance data with spiked workload

	JD Edwards Enterprise One 500 users	**JD Edwards Enterprise Learning Management 600 users**	**Trade3 1,000 users**	**Samba File Serving 25 users**
Standalone 2-way, fixed partition	Response time: 0.7 seconds Processor: 69%	Response time: 3.5 seconds Processor: 63%	Transactions/sec: 66 Processor: 57%	Throughput: 41 MB/sec Processor: 40%
Standalone Batch in Fixed Partition	Processor: 50% Wall: 13:35 minutes	no activity	no activity	no activity
Concurrent Fixed Partition with Batch	Response time: 0.4 seconds Processor: 90% Wall: 33:26 minutes	Response time: 3.6 seconds Processor: 64%	Transactions/sec: 66 Processor: 59%	Throughput: 39 MB/sec Processor: 41%
Concurrent Shared Partition with Batch	Response time: 0.4 seconds Processor: 118% Wall 26:43 minutes	Response time: 3.7 seconds Processor: 70%	Transactions/sec: 66 Processor: 65%	Throughput: 38 MB/sec Processor: 43%

In this situation, JD Edwards Enterprise One initially runs on a fixed partition and has a maximum of two processors at its disposal. Server jobs that handle "interactive" HTML requests always run at a higher priority than batch jobs. So although the batch job could use 50% of the processor's resources, it has only about 20% of the processor's resources at its disposal when running the spiked workload. As a result, the wall time (time measured by a wall clock or wristwatch, as opposed to time measured in microprocessor clock pulses or cycles), increases from 13:35 minutes to 33:26 minutes, an increase of about 250%.

By changing the system from fixed partitions to shared partitions and dynamically reallocating resources from micro-partitions doing less work, the wall time is reduced to 26:43 minutes, an improvement of about 20% with minimal disruption to the other partitions.

In this case, prior to spiking, the partitions have average processor usage rates ranging from 43% to 73%. This is higher than expected for normal client situations, so it is possible that clients might see greater benefits by using shared partitions.

It is important to note that this is a synthetic test using simulated applications. Clients should validate their own applications prior to using this technique in a production environment.

Comparison to other published numbers

Table 6-4 provides a comparative guide to how the 2-way partition benchmark results compare to other published results. These new results are not certified. Therefore, you should use them only as a general guide to help compare diverse architectures and platforms.

Table 6-4 Comparisons of various benchmarks on different platforms

Workload	IBM @server i5 environment	IBM @server i5 metrics	Comparative environment	Comparative metrics
JD Edwards Enterprise One 8.9	IBM @server i5 2-way partition, 1.65 GHz simultaneous multi-threading processors, 32 GB of memory, 45 35 GB disks, i5/OS V5R3, DB2 UDB, WebSphere 5.02	500 users response time: 0.4 seconds Processor: 73%	IBM POWER4™ usage, 3-way partition, 1.1 GHz processors, 8 GB of memory, 15 35 GB disks, OS/400 V5R2, DB2 UDB, WebSphere 5.02	300 users Response time: 0.6 seconds Processor: 69%
JD Edwards Enterprise Learning Management 8.81	IBM @server i5 2-way partition, 1.65 GHz simultaneous multi-threading processors, 32 GB of memory, 25 35 GB disks, AIX 5L V5.3, DB2 V8 FP4, WebSphere 4.03	60 users response times: Login: 1.25 seconds Retrieve: 1.50 seconds Update: 0.79 seconds Processor: 70% (Web server, application and database)	Database: 4-way HP RX5670 server, Intel Itanium® 2 processors at 1 GHz, 32 GB of memory, 15 36 GB disks, Application: 16-way HP RP8400 8700 server, 875 MHz processors, 64 GB of memory, 144 GB disk, Web server: 2-way RP2470 8700 server, 750 MHz processors, 8 GB of memory, 108 GB disk, HP-UNIX 11, Oracle 9i, WebLogic 6.1	600 users Response times: Login: 1.72 seconds Retrieve: 1.99 seconds Update: 0.91 seconds Processor: 15% database, 30% Web server, 33% application 1,200 users Response times: Login: 2.25 seconds Retrieve: 3.00 seconds Update: 1.07 seconds Processor: 30% database, 47% Web server, 75% application
Trade3	IBM @server i5 2-way partition, 1.65 GHz simultaneous multi-threading processors, 32 GB of memory, 45 35 GB disks, i5/OS V5R3, DB2 UDB, WebSphere 5.1	1000 users Transactions/sec: 66 Processor: 65%	1-way IBM @server i5 520, 1.65 GHz simultaneous multi-threading processor, 8 GB of memory, 20 35 GB disks, i5/OS V5R3, DB2 UDB, WebSphere 5.1	500 users Transactions/sec: 32 Processor: 79%
Samba File Server Version 3.0.6 Linux firewall	IBM @server i5 2-way partition, 1.65 GHz simultaneous multi-threading processors, 16 GB of memory, 15 35 GB disks, SUSE LINUX Enterprise Server 9	25 users throughput: 38 MB/sec Processor: 38% 1,025 users throughput: 39 MB/sec Processor: 5%	IBM @server zSeries® Z900 2064 model 116, 16-way, 64 GB memory, 1.6TB disk, SUSE LINUX Enterprise Server 7	50–70 users throughput: 106 MB/sec

6.5 Summary

The Virtualization Grand Slam benchmark demonstrates how applications can run on i5/OS, AIX 5L, Linux and Windows platforms on a single IBM @server i5 server without compromising performance or manageability. By using shared processor partitions, processor resources can be balanced across partitions to minimize the impact of sudden increases in application workload.

The POWER5 architecture provides new server consolidation, performance and virtualization opportunities for clients using IBM @server i5 systems. In addition, IBM Virtualization Engine Console software creates a common interface to monitor disparate workloads, taking advantage of existing products such as Management Central and IBM Director Multiplatform Agent.

7

Switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment

High availability solutions come in various shapes and forms. One of the most straightforward to implement and to use is generically called a *switched disk* or *switchable device* topology. With IBM @server iSeries, this solution type is referred to as switchable independent auxiliary storage pools (IASPs). This chapter, though technical in nature, demonstrates how straightforward it is to set up and operate a switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment.

© Copyright IBM Corp. 2005. All rights reserved.

7.1 Hardware environment architecture

The hardware consists of two iSeries servers and one DASD tower, all connected in an HSL loop configuration. The two iSeries servers are configured in a two-node cluster with one system functioning as the primary node and the other as the backup node. The DASD tower is configured to serve as a switchable IASP, to be used by either system. A cluster resource group (CRG) is configured between the two nodes and used to associate an IP address with the switchable IASP. This IP address is called a takeover IP, because when control of the IASP is switched from one node to the other, the IP address is switched as well. In other words, the IP address is taken over by the node that is controlling the IASP, hence the name takeover IP.

Note: For this project, the numeric IP address used to configure the takeover IP interface is resolved to a specific name using the network Domain Name Server (DNS). When the takeover IP is referred to throughout this chapter, it is the IP address name, not the numeric IP address, that is being referenced.

Figure 7-1 illustrates the hardware environment configuration.

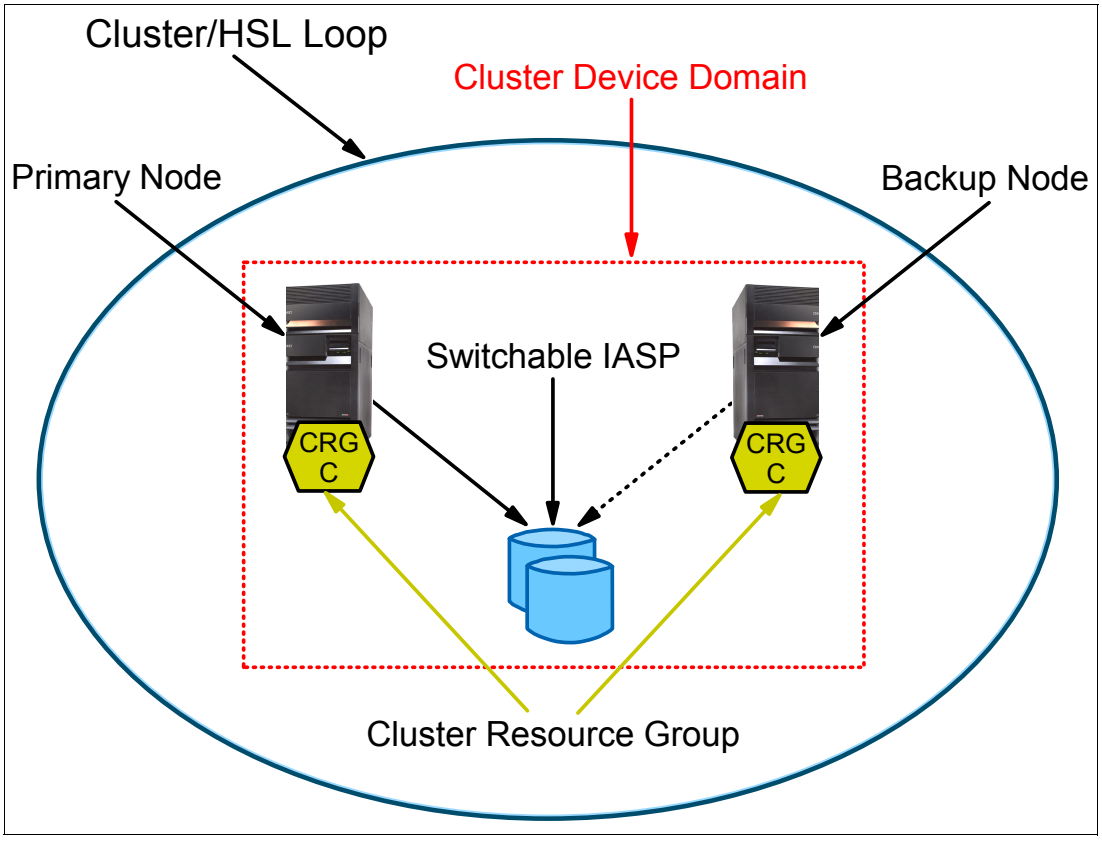

Figure 7-1 Hardware environment

7.2 Software environment architecture

The JD Edwards EnterpriseOne ERP 8.0 software environment architecture used in this project consists of an application server, a database, and a Java Application Server (JAS). The application server and the database are first installed on the primary node. After the installation is complete, the following objects are moved from the system ASP of the primary node to the IASP:

- All JD Edwards EnterpriseOne ERP 8.0 application server directories in the Integrated File System (IFS)
- All JD Edwards EnterpriseOne ERP 8.0 application server libraries, except the B7334SYS library
- All JD Edwards EnterpriseOne ERP 8.0 database libraries

Because the IASP is configured to be switchable, all the libraries and directories in the IASP move from one iSeries server to the other when control of the IASP is switched from one node to the other.

The application server library B7334SYS remains in the system ASP of the primary node. A copy is placed in the system ASP of the backup node. Therefore, you can start JD Edwards EnterpriseOne services on the node that is currently controlling the IASP (see Figure 7-2 on page 88).

Note: All JD Edwards EnterpriseOne user profiles that exist on the primary node system must also be created on the backup node system. Thus, any user profile that uses the JD Edwards EnterpriseOne application for any reason must exist on both the primary and backup system nodes.

The JAS location is not mentioned because this configuration works as long as the following two conditions are true:

- The HTML clients are able to establish a network connection to the JAS.
- The JAS is able to establish a network connection to the JD Edwards EnterpriseOne Application Server by using the takeover IP.

Figure 7-2 on page 88 illustrates the software environment configuration.

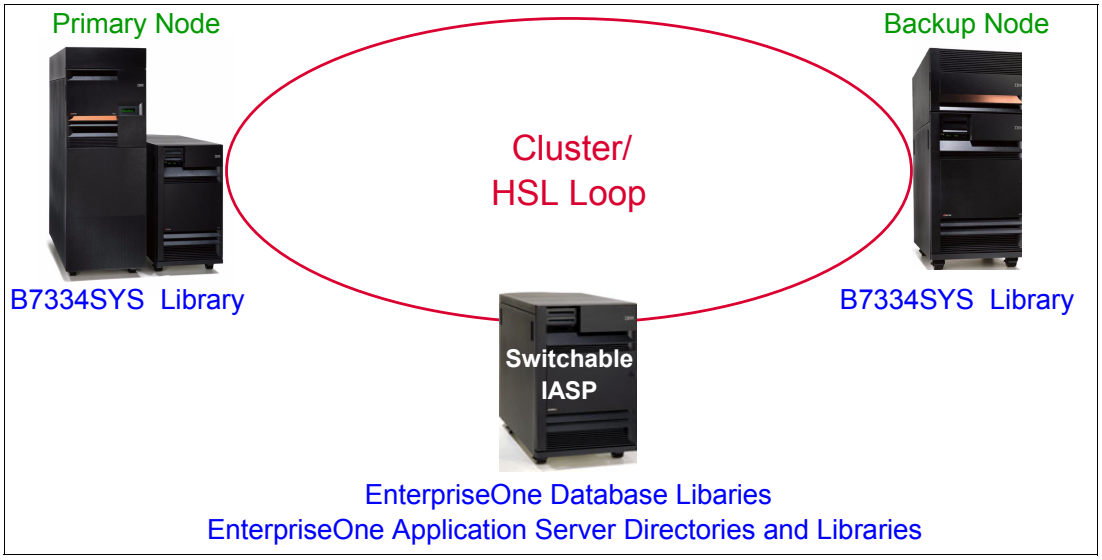

Figure 7-2 Software environment

7.3 Configuring the IASP for JD Edwards EnterpriseOne

The focus of this section is the JD Edwards EnterpriseOne application and the steps needed to implement it in a switchable IASP environment. The steps necessary to create the cluster, the IASP, and the CRG are not documented here. With that said, you must address one important IASP configuration issue to insure that the JD Edwards EnterpriseOne application functions correctly: the name of the Relational Database Directory Entry associated with the IASP must match the name that is used for the takeover IP.

To understand the concepts of Relational Database Directory (RDB entries) and associating the entries to an IASP, see Appendix C, “Additional information for switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment” on page 145.

Creating the IASP with the same name as the takeover IP name is an easy way to insure that its RDB entry name is the same as the takeover IP name. This is because when an IASP is created, the system adds an RDB entry for the IASP with the same name as the IASP. However, if the IASP is already created with a name that does not match the name of the takeover IP, a configuration change is required.

There are two possible configuration changes to make the name of the takeover IP match the name of the RDB entry associated with the IASP:

- Change the name of the takeover IP to match the name of the RDB entry that is associated with the IASP. To accomplish this, contact your network administrator.
- Change the name of the RDB entry that is associated with the IASP to match the name of the takeover IP. To accomplish this:

  a. Enter the following command from the green screen to create an RDB entry with the same name as the takeover IP:

  ```
  ADDRDBDIRE RDB(name of takeover IP) RMTLOCNAME(LOOPBACK *IP)
  ```

  b. Enter the following command from the green screen to associate the RDB entry that was just created with the IASP:

  ```
  CHGDEVASP DEVD(name of IASP) RDB(name of takeover IP)
  ```

7.4 Moving JD Edwards EnterpriseOne libraries to the IASP

All of the following libraries, with the exception of the B7334SYS library, are moved to the IASP. The B7334SYS library is not moved to the IASP because it contains objects that are not supported in the IASP.

- B7334SYS
- COPD7334
- DD7334
- OL7334
- OWJRNL
- PD7334
- PD7334DNT
- PD7334FA
- PRODCTL
- PRODDTA
- SYS7334
- SVM7334

It is not possible to journal a library from the system ASP to a journal in the IASP and vice versa. Therefore, when moving database libraries that are journaled to the IASP, you must move the journals and journal receives for these libraries as well. In other words, a library and its journal must both be either in the system ASP or in an IASP.

Note: Complete a full system save to tape before attempting the instructions given in this section.

Follow these steps to move the JD Edwards EnterpriseOne libraries to the IASP:

1. Save the libraries to save files.
2. Delete the libraries from system space (the system ASP).
3. Restore database libraries to the IASP.

These steps are described in detail in the following sections.

7.4.1 Saving the libraries into save files

Use the following commands from the green screen to create a save file for each of the libraries, except the B7334SYS library, and to save each library into the appropriate save file:

```
CRTSAVF FILE(Name of library to containing save file / Save File Name)
SAVLIB LIB(Name of library being saved) DEV(*SAVF) +
SAVF(Name of library containing save file /Save File Name) ACCPTH(*YES) + DTACPR(*YES)
```

7.4.2 Deleting the libraries from the system ASP

Note: Verify that all the libraries have been fully saved into save files before proceeding with the instructions in this section. You should also do a full system save before beginning.

It is necessary to delete all the original libraries from the system ASP because the same library name cannot exist in the system ASP and the independent ASP at the same time. Before any libraries can be restored to the independent ASP, remove them from the system ASP. Delete all the libraries from the system ASP, except for the B7334SYS library.

Use the following command from the green screen to delete the libraries from the system ASP:

```
DLTLIB LIB(Name of library to be deleted)
```

7.4.3 Restoring the database libraries to the IASP

After all the libraries are removed from the system ASP, you can restored them to the IASP from the save files. Restore all the libraries, except the B7334SYS library, to the IASP.

Use the following command from the green screen to restore the libraries to the IASP:

```
RSTLIB SAVLIB(Name of library being restored) DEV(*SAVF)  +
SAVF(Name of library where save file is / Save File Name) +  RSTASP(Name of the IASP)
```

7.5 Moving the JD Edwards EnterpriseOne IFS directories to the IASP

When an IASP is created, an IFS directory is created in the system's root directory with the same name as the IASP. Any IFS objects that reside in this directory are actually in the IASP's IFS directory, not the system ASP's IFS directory. To move IFS objects to the IASP, simply copy them into the IASP's IFS directory and then delete them from the system ASP's IFS directory.

The JD Edwards EnterpriseOne environment used in the project were:

- B7334SYS
- Jdeb7334
- PD7334
- OneWorld

All the IFS directories were moved to the IASP directory. To move directories to the IASP:

1. Copy the directories to the IASP's IFS directory and delete them from the system ASP's IFS directory.
2. Create symbolic links in the system ASP's IFS directory on both nodes which point to the directories' new locations in the IASP's IFS directory.

7.5.1 Coping the directories to the IASP

Use iSeries Navigator to copy the directories to the IASP directory. Click the plus sign (+) beside `My Connections` to expand the list of systems that are available. Then click the plus sign (+) beside the name of the system where the IASP resides. Next, click the plus sign (+) beside `File Systems` to expand the directory list for that system. The iSeries Navigator screen should now look similar to the one in Figure 7-3 on page 91.

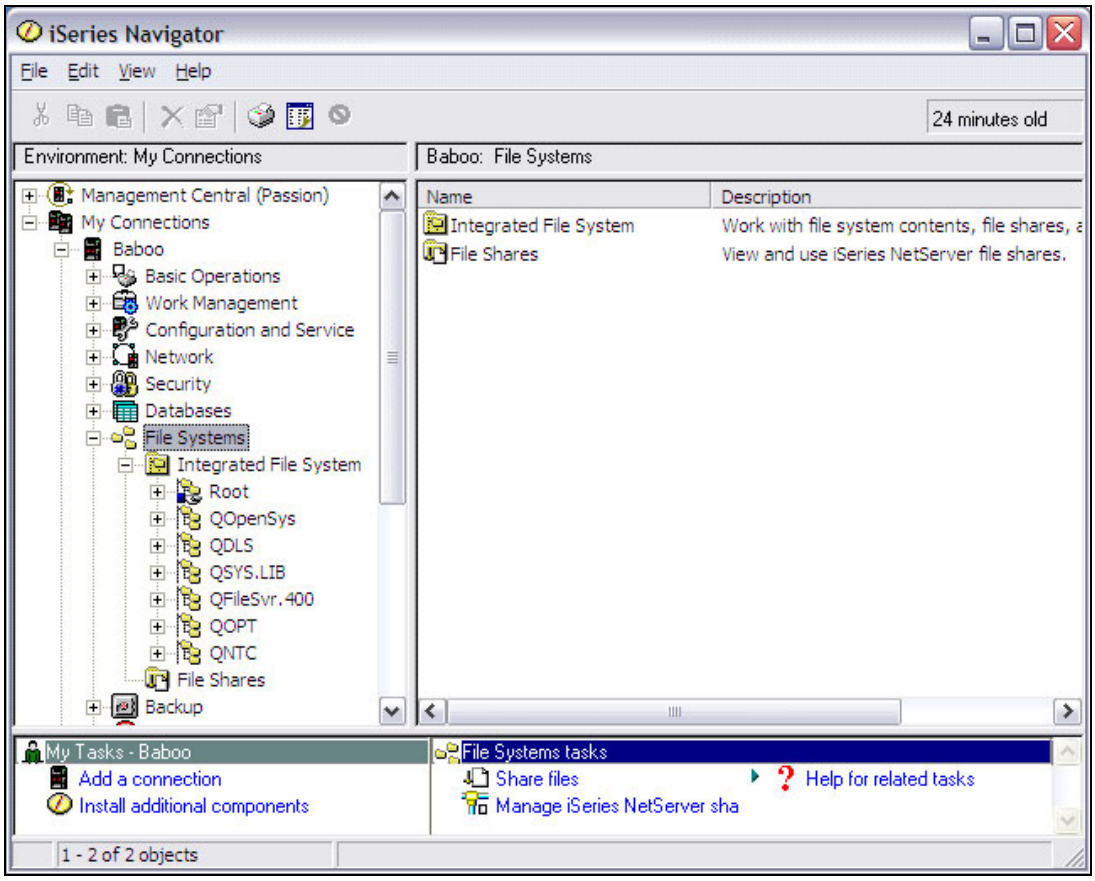

Figure 7-3 iSeries Navigator copy directory

The Root directory is where the IFS directory for the IASP resides. So, click the plus sign (+) beside the Root directory. When all the directories under Root are visible, find the directory with the same name as the IASP (see Figure 7-4 on page 92). This is the directory into which you must copy all the JD Edwards EnterpriseOne directories.

Select the name of the directory to be copied, and use the copy and paste commands found on the edit menu. When the copying is finished, delete the directories from the system ASP's IFS directory. Select the name of the directory to be deleted and select **Delete**.

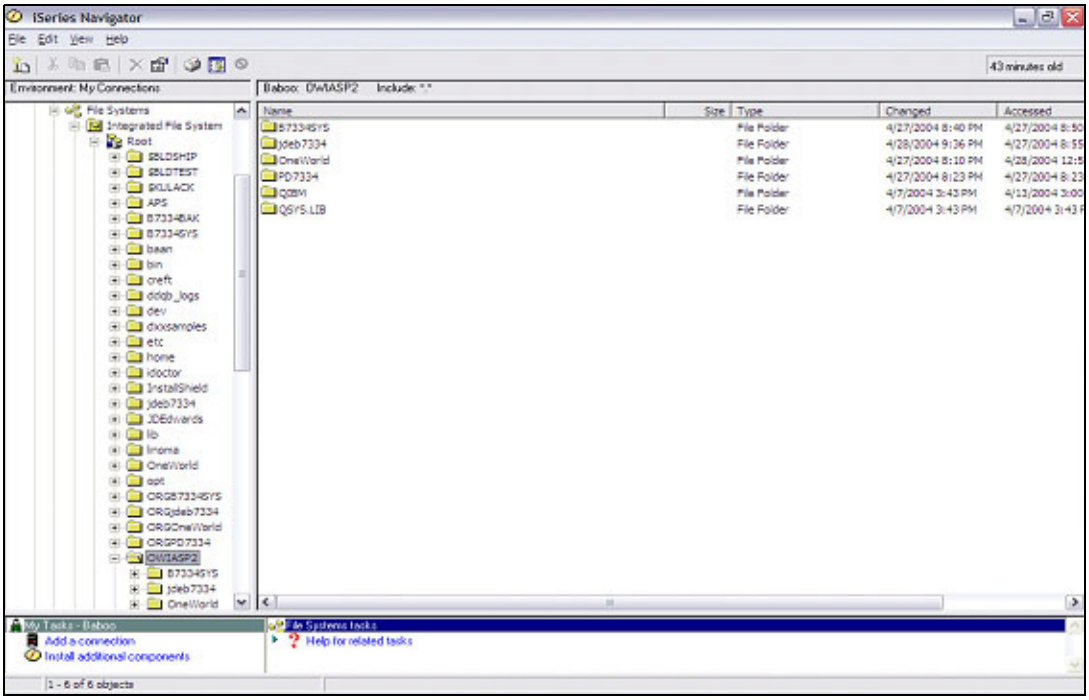

Figure 7-4 iSeries Navigator delete directory

7.5.2 Creating symbolic links

A symbolic link must be created for every IFS directory that is moved from the system ASP's IFS directory to the IASP's IFS directory. A symbolic link with the same name as the directory is placed in the directory's original location. It points to its new location in the IASP's IFS directory. This is done so that the JD Edwards EnterpriseOne application is able to find the directories in their new location in the IASP's IFS directory.

Use the following command from the green screen to create a symbolic link for each of the directories moved to the IASP's IFS directory:

```
ADDLNK OBJ('IASP Name/Directory Name') NEWLNK('Directory Name')
```

Add a symbolic link to the system ASP's IFS directory on both the primary and backup node for each directory that is moved to the IASP's IFS directory. Use the following command from the green screen to view a symbolic link that is created:

```
WRKLNK DETAIL(*EXTENDED)
```

A screen similar to Figure 7-5 on page 93 should appear.

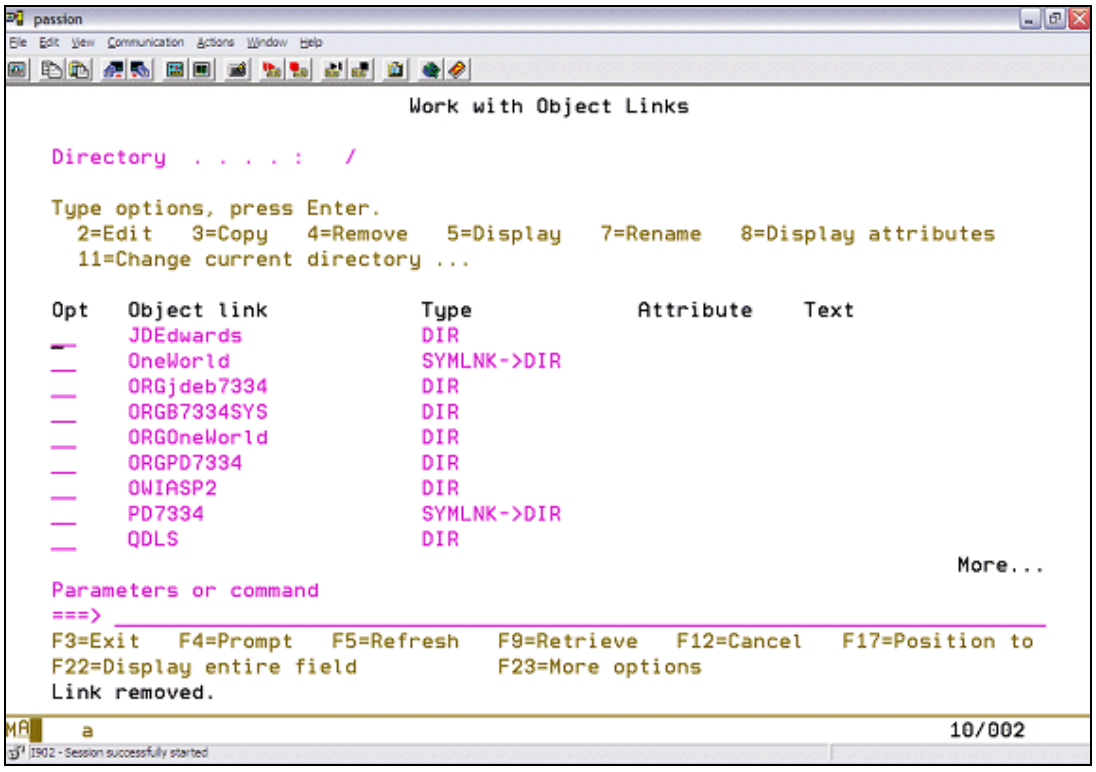

Figure 7-5 Working with Object Links

7.6 Configuring JD Edwards EnterpriseOne data sources and OCM tables

Note: The following instructions are written for JD Edwards EnterpriseOne ERP 8.0. It is possible that changes are made to the libraries and tables mentioned here in later releases of JD Edwards EnterpriseOne. If you are using a JD Edwards EnterpriseOne release later than JD Edwards EnterpriseOne ERP 8.0, consult JD Edwards EnterpriseOne documentation before proceeding with these instructions.

The JD Edwards EnterpriseOne data sources table (named the F98611 table) defines how the application should access either data or servers. It is found in both the SYS7334 and SVM7334 libraries. This table works in conjunction with the JD Edwards EnterpriseOne Object Configuration Manager or OCM table. The OCM table (named the F986101 table) defines where the database tables are located, where business functions run, and where Universal Batch Engines (UBEs) run. It is also found in the SYS7334 and SVM7334 libraries.

Because these tables are installed on the primary node, they contain the primary node system's host name. Edit these tables to reflect the takeover IP name in order for JD Edwards EnterpriseOne to function correctly in a clustered IASP environment. This is important so the mapping in these tables is not tied to either system's host name but instead work regardless of which system has control of the IASP.

You use the following SQL commands to edit the tables:

- Edit the F98611 table in the SVM7334 library

```
Takeover IP = name of takeover IP address.
PnodeName = host name of primary node system.

UPDATE svm7334.F98611 SET (OMDATP) = ('Takeover IP - B7334 Server Map') where OMDATP =
'PNodeName - B7334 Server Map'   ;
UPDATE  svm7334.F98611 SET (OMDATP) = ('Takeover IP') where OMDATP = 'PNodeName'   ;
UPDATE  svm7334.F98611 SET (OMSRVR) = ('Takeover IP') where OMSRVR = 'PNodeName'   ;
UPDATE  svm7334.F98611 SET (OMDATB) = ('Takeover IP - B7334 Server Map') where OMDATB =
'PNodeName - B7334 Server Map'   ;
UPDATE  svm7334.F98611 SET (OMLL) = ('Takeover IP') where OMLL = 'PNodeName'   ;
```

- Edit the F986101 table in the SVM7334 library

```
Takeover IP = name of takeover IP address.
PnodeName = host name of primary node system.

UPDATE svm7334.F986101 SET (OMDATP) = ('Takeover IP') where OMDATP = 'PNodeName'   ;
```

- Edit the F98611 table in the SYS7334 library

```
Takeover IP = name of takeover IP address.
PnodeName = host name of primary node system.

UPDATE sys7334.F98611 SET (OMDATP) = ('Takeover IP - B7334 Server Map') where OMDATP =
'PNodeName - B7334 Server Map'   ;
UPDATE  sys7334.F98611 SET (OMDATP) = ('Takeover IP') where OMDATP = 'PNodeName'   ;
UPDATE  sys7334.F98611 SET (OMDATP) = ('Takeover IP - Logic') where OMDATP = 'PNodeName
- Logic'   ;
UPDATE  sys7334.F98611 SET (OMSRVR) = ('Takeover IP') where OMSRVR = 'PNodeName'   ;
UPDATE  sys7334.F98611 SET (OMDATB) = ('Takeover IP - B7334 Server Map') where OMDATB =
'PNodeName - B7334 Server Map'   ;
UPDATE  sys7334.F98611 SET (OMLL) = ('Takeover IP') where OMLL = 'PNodeName'   ;
```

- Edit the F986101 table in the SYS7334 library

```
Takeover IP = name of takeover IP address.
PnodeName = host name of primary node system.

UPDATE sys7334.F986101 SET (OMDATP) = ('Takeover IP') where OMDATP = 'PNodeName'   ;
UPDATE  sys7334.F986101 SET (OMDATP) = ('Takeover IP - Logic') where OMDATP = 'PNodeName
- Logic'   ;
```

Note: The SQL commands that are given use the iSeries Navigator's SQL environment. Change the syntax so that the commands run correctly in other SQL environments.

7.7 Configuring JD Edwards EnterpriseOne user profiles

Most of the JD Edwards EnterpriseOne objects are moved to the IASP. Therefore, be sure that the library namespace connected to by the JD Edwards EnterpriseOne users is the IASP's library namespace. When a user logs on to an iSeries, by default they are connected to the system ASP library name space. Edit the user's job description so that it points to the IASP. This ensures that a user at log in connects to the IASP's library name space.

Edit the job description in all JD Edwards EnterpriseOne user profiles on both nodes to point to the IASP. Any job description in any profile, on either node, that uses the JD Edwards EnterpriseOne application for any reason must point to the IASP.

Note: Because all JD Edwards EnterpriseOne user profiles must exist on both the primary and backup system nodes, complete the instructions in this section for both nodes.

7.7.1 Creating a job description

The JD Edwards EnterpriseOne user profiles of the systems used for this project utilize a job description named ONEWORLD. A copy of the ONEWORLD job description is made and then edited.

Use the following command from the green screen to find the job description that is used by the JD Edwards EnterpriseOne user profiles on your system:

```
DSPUSRPRF USRPRF(EnterpriseOne user name)
```

The Display User Profile - Basic screen appears as shown in Figure 7-6.

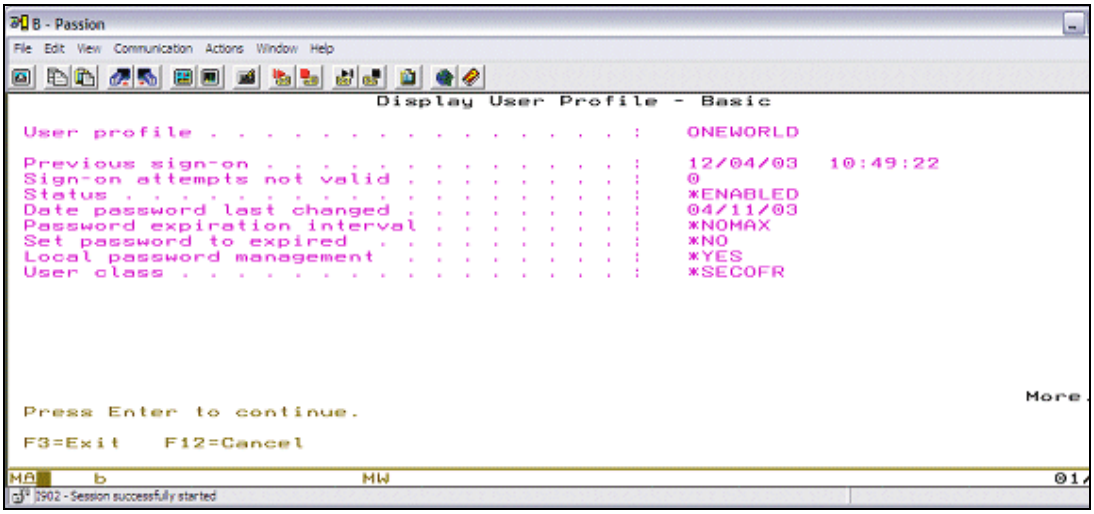

Figure 7-6 Display User Profile - Basic

Page down until you see the line that is labeled Job description a shown in Figure 7-7 on page 96. This is the job description that the JD Edwards EnterpriseOne user profiles are using.

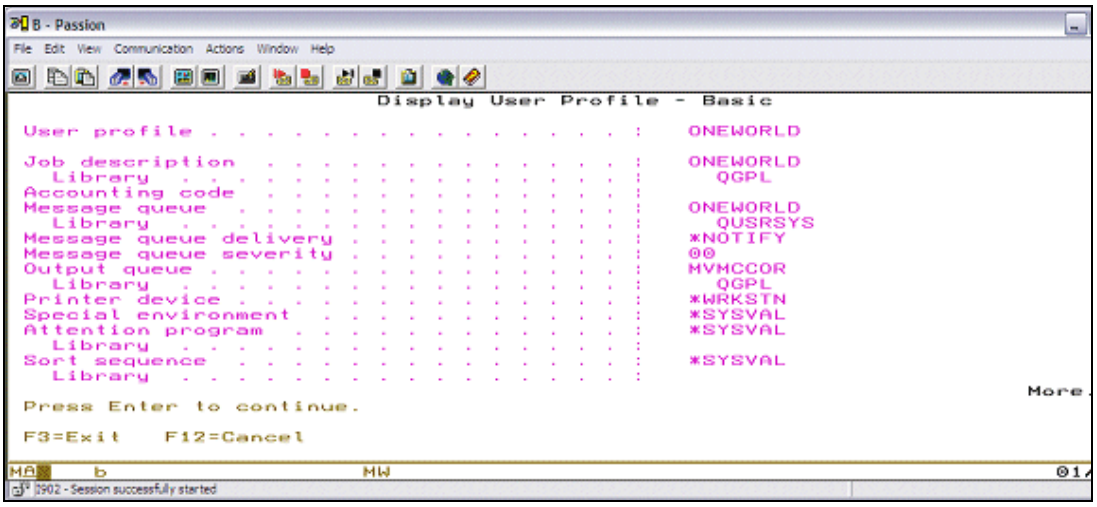

Figure 7-7 Display User Profile - Basic, continued

You should make a copy of the job description that is currently used by the JD Edwards EnterpriseOne users on your system. Use the following command from the green screen to copy the job description:

```
CRTDUPOBJ OBJ(name of EnterpriseOne User's jobd) FROMLIB(library where jobd resides)
OBJTYPE(*JOBD) NEWOBJ(new jobd name)
```

Note: Make sure the new job description is created in a library in the system ASP on both node, not the IASP. The Create Duplicate Object (**CRTDUPOBJ**) command, by default, places the new object in the same library as the object being copied.

The new job description points to the system ASP library namespace. Use the following command from the green screen to edit the new job description so that it points to the IASP's library namespace:

```
CHGJOBD JOBD(library where jobd resides / new jobd name) INLASPGRP(IASP name)
```

7.7.2 Adding the new job description to user profiles

Now that the job description is created, edit all JD Edwards EnterpriseOne user profiles on both nodes to use the new job description. In this way, the user profiles defaults to the IASP when a user logs on.

Use the following command from the green screen to edit the JD Edwards EnterpriseOne user profiles:

```
CHGUSRPRF USRPRF(EnterpriseOne user name) JOBD(library where jobd resides / new jobd)
```

Note: Change all JD Edwards EnterpriseOne user profiles, on both nodes, to use the new job description which points to the IASP.

Use the following command from the green screen to edit the job description to default to the IASP once the new job description is created:

```
CHGJOBD JOBD(library where jobd resides / new jobd name) INLASPGRP(IASP name)
```

7.8 Editing the JD Edwards EnterpriseOne enterprise server's jde.ini file

The JD Edwards EnterpriseOne Application Server uses the settings in its jde.ini file to establish its connectivity to various database sources and servers. In order for JD Edwards EnterpriseOne to function correctly in a clustered IASP environment the jde.ini file needs to be edited. This is to insure that JD Edwards EnterpriseOne establishes its connections based on the takeover IP and not any system host names.

The jde.ini file resides in the B7334SYS library and a copy of this library is in the system ASP on both the primary and back up node systems. Therefore, you should execute the instructions in this section on both the primary and backup node systems.

When the OneWorld Server Administration Workbench screen appears, sign on to the system as `ONEWORLD` and enter the **`SAW`** command to edit the Enterprise Server's jde.ini file (see Figure 7-8). Choose option 5 (`Work with Server INI file`).

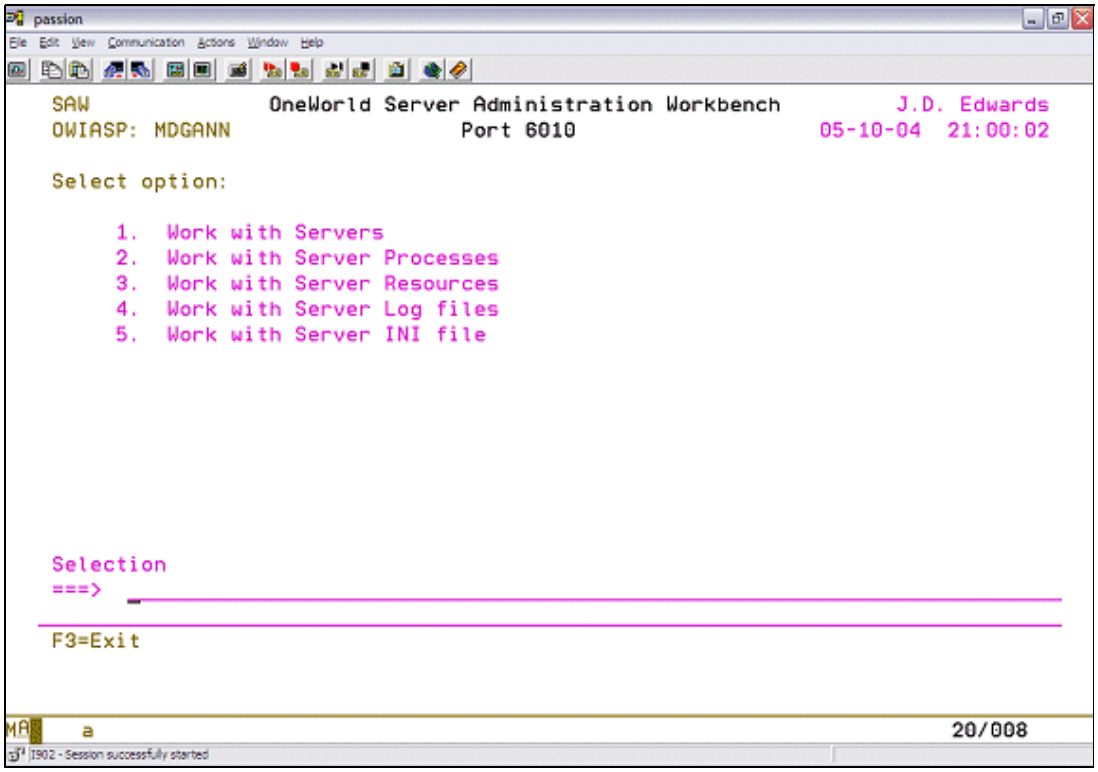

Figure 7-8 Work with Server INI file

Add the following section with a label of [CLUSTER] to the jde.ini file. The Cluster section is only used for clustered environments and is not in the jde.ini file by default.

```
[CLUSTER]
PrimaryNode=Takeover IP
```

Edit the settings in the section labeled [DB SYSTEM SETTINGS] as follows. The statements are already in the jde.ini file.

```
[DB SYSTEM SETTINGS]
Base Datasource=Takeover IP - B7334 Server Map
Server=Takeover IP
Database=Takeover IP - B7334 Server Map
```

Edit the settings in the section labeled [SECURITY] as follows. The statements are already in the jde.ini file.

```
[SECURITY]
SecurityServer=Takeover IP
```

Save the jde.ini file after making the changes.

7.9 Adding a host table entry for the takeover IP

The JD Edwards EnterpriseOne Application Server uses TCP/IP host table entries to establish connectivity when running in a clustered IASP environment. In order for JD Edwards EnterpriseOne to function properly in a clustered IASP environment, host table entries for the takeover IP must be added to both the primary and backup node systems.

Use the following command from the green screen to add a host table entry to the system:

```
ADDTCPHTE INTNETADR('numeric take over IP address') HOSTNAME(('takeover IP Name')
('takeover IP Name.domain name.com'))
```

Remember to add a host table entry for the takeover IP on both the primary and backup systems.

7.10 Configuring JD Edwards EnterpriseOne fat clients

In order for the fat client machines to work correctly in an IASP environment there are two configuration connections issues that need to be addressed. The fat client ODBC connections and the fat client network connections need to be configured to connect to the server using the takeover IP address, rather than a system host name. This is to insure that the fat clients work regardless of whichever node is currently controlling the IASP.

Repeat the instructions in this section for all JD Edwards EnterpriseOne data sources that exist on all fat client machines.

7.10.1 Changing the ODBC connections

Launch the ODBC Data Source Administrator to change the ODBC settings on a fat client machine by selecting **Start** → **Control Panel** → **Administrative Tools** → **Data Sources (ODBC)**. Double-click **Data Sources (ODBC)** and the ODBC Data Source Administrator dialog box opens (see Figure 7-9).

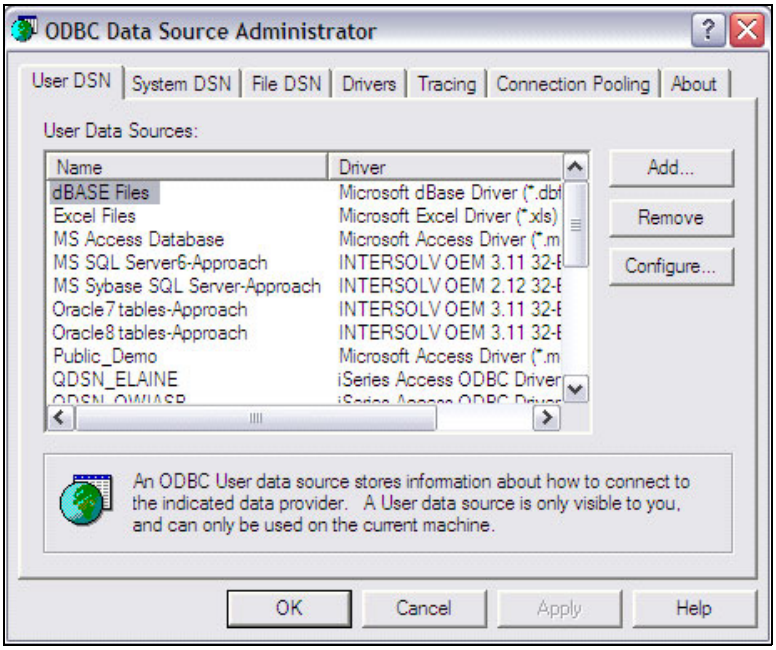

Figure 7-9 ODBC Data Source Administrator

When the ODBC Data Source Administrator dialog box opens, select the System DSN tab, as shown in Figure 7-10. On the System DSN tab, click the Data Source Name to highlight it.

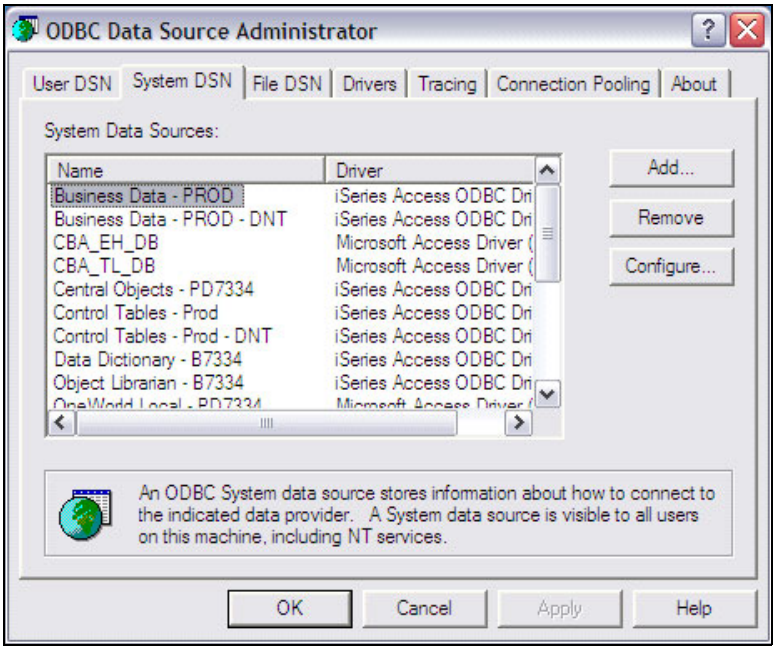

Figure 7-10 ODBC Data Source Administrator, continued

Then, click **Configure** so that the iSeries Access for Windows ODBC Setup dialog appears (see Figure 7-11).

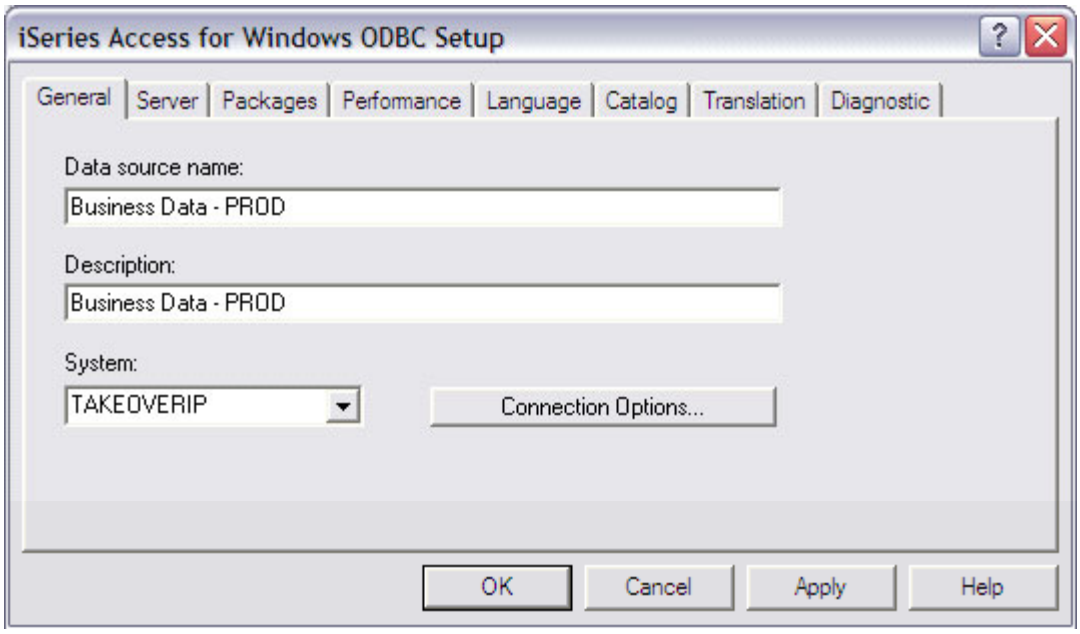

Figure 7-11 iSeries Access for Windows ODBC Setup

When the iSeries Access for Windows ODBC Setup dialog appears, type the takeover IP name in the System pull-down text box. After you enter the takeover IP name, click **Apply** and then click **OK** to close the iSeries Access for Windows ODBC Setup dialog.

7.10.2 Network connections

Edit the jde.ini file on the fat client machine to configure the fat client to make its network connections use the takeover IP address. Locate the jde.ini file on the fat client machine. It is usually found in the C:\WINDOWS\JDE.INI directory.

After you locate the file, open it using any text editor that you prefer.

- Change the following statements in the `[DB SYSTEM SETTINGS]` section:

```
Server=TAKEOVERIP
Library List= TAKEOVERIP
```

- Change the following statements in the `[SECURITY]` section:

```
SecurityServer=TAKEOVERIP
```

Save the file after making the changes.

You can find information about the contents of the jde.ini file in Appendix B, “JD Edwards EnterpriseOne 8.9 test configuration files” on page 125.

7.11 Configuring the JD Edwards EnterpriseOne Java Application Server

To insure that the JD Edwards EnterpriseOne Java Application Server functions correctly in an IASP environment, one configuration issue needs to be addressed. You must configure the network connections for the JAS to connect to the server that is using the takeover IP rather than a system host name. This is to insure that JAS functions correctly regardless of the node that is currently controlling the IASP.

To configure the Java Application Server to make network connections using the takeover IP, it is necessary to edit the jas.ini file on the system where the Java Application Server resides.

Use the following command from the green screen to edit the jas.ini file:

```
wrklnk  "/JDEdwards/JAS/EA_JDEdwards_1.ear/webclient.war/ WEB-INF/jas.ini"
```

When the Work With Object Link screen appears, type a 2 next to file name and press Enter. This opens the file in edit mode as shown in Figure 7-12.

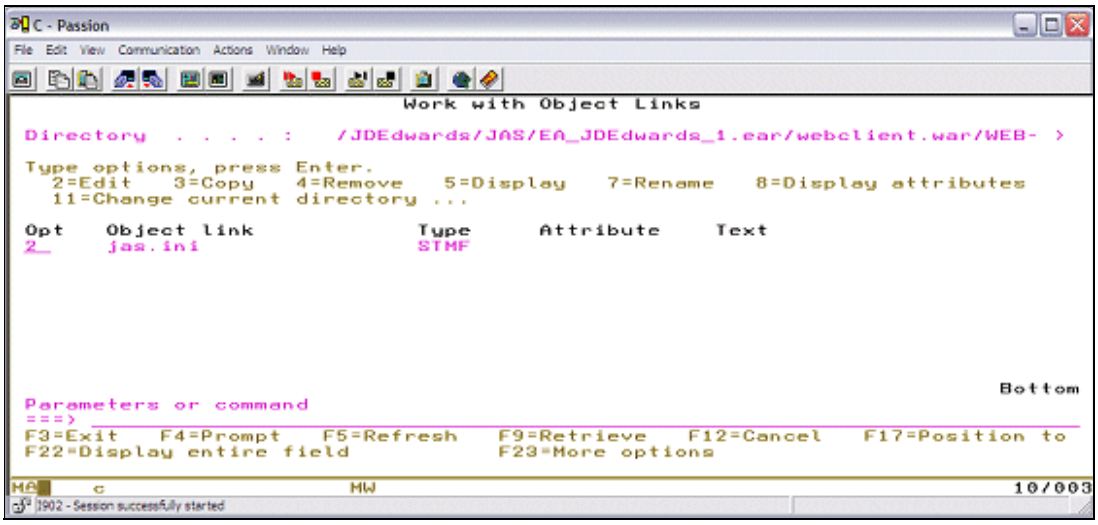

Figure 7-12 Work with Object Links

After the jas.ini file is opened in edit mode, edit the following sections.

- Change the following statement in the [DB SYSTEM SETTINGS] section:

  ```
  Server=Takeover IP
  ```

- Change the following statement in the [SECURITY] section:

  ```
  SecurityServer=Takeover IP
  ```

Save the file after making the changes.

7.12 Switching between nodes

Note: If running a JD Edwards EnterpriseOne Java Application Server, stop the JAS instance and restart it after you have completed the steps outlined in this section. You should also stop and restart the JD Edwards EnterpriseOne HTTP server.

To switch from the primary node system to the backup node system:

1. Stop the JD Edwards EnterpriseOne Application Sever on the primary node system.
2. Log on to the primary node system as `ONEWORLD` and enter the following command:

   ```
   endnet
   ```

   The JD Edwards EnterpriseOne Application Server on this system shuts down.
3. Switch control of the IASP from the primary system node to the backup system node. The steps to achieve this are documented in *IBM @server iSeries Independent ASPs: A Guide to Moving Applications to IASPs,* SG24-6802.
4. Start the JD Edwards EnterpriseOne Application Sever on the backup node system.

   Log on to backup node system as `ONEWORLD` and enter the following command:

   ```
   strnet
   ```

   The JD Edwards EnterpriseOne Application Server on this system now starts.

After you have completed these steps, the JD Edwards EnterpriseOne environment is fully operational. The fat clients and HTML clients are able to function normally, unaware that the JD Edwards EnterpriseOne Application Server is running on the backup system.

7.12.1 Performance considerations when switching nodes

There are several factors that can affect how long it takes the IASP to switch from one node to the other. Two recommendations came out of this project:

- Keep the number of database objects in the system ASP as low as possible. In fact, if possible, place all database objects in the IASP.
- Make sure that user profiles that own objects in the IASP have the same user ID numbers (UIDs) and group ID numbers (GIDs) on both the primary and backup node systems.

  The easiest way to synchronize UIDs and GIDs on both nodes is to use iSeries Navigator. The steps to synchronize across both nodes using iSeries Navigator are documented in “Synchronizing user profiles across nodes” on page 148.

7.13 Further resources

Developing and operating a JD Edwards EnterpriseOne IASP solution is a relatively straightforward procedure. For educational opportunities and implementation assistance, visit the iSeries Technology Center Web site at:

`http://www.ibm.com/servers/eserver/iseries/service/itc/index.html`

Additional information about iSeries High Availability solutions can also be found on the High Availability Web site at:

`http://www.ibm.com/servers/eserver/iseries/ha/`

8

Storage area network technology

This chapter discusses storage area network (SAN) technology and how it relates to iSeries servers that have implemented JD Edwards EnterpriseOne (formerly J.D. Edwards OneWorld) or JD Edwards World software solutions. The certain SAN benefits of simplified storage management, disk virtualization, backup, disaster recovery, and performance — to name a few — are available on the iSeries server, often driven by the storage consolidation of OS/400, Windows, and UNIX platforms. IBM, in particular the iSeries servers, have extended system support of SAN technology.

This chapter allows JD Edwards users to better understand what the SAN technologies are and how those technologies are applicable to JD Edwards environments. This chapter is not intended to serve as a detailed sizing or implementation guide. See “Related publications” on page 153 for a number of SAN implementation, SAN design, and disk subsystem planning guides.

Note: All references in this chapter to Oracle customers imply that they are also running their JD Edwards EnterpriseOne or JD Edwards World solution set on an IBM @server i5 or iSeries server.

© Copyright IBM Corp. 2005. All rights reserved.

8.1 Overview of SAN technology

A SAN is a dedicated, centrally managed, secure information infrastructure which enables any-to-any interconnection of servers and storage systems via the fibre channel protocol. Simplistically, a SAN is similar to a local area network (LAN) and a wide area network (WAN) in its architecture. However, SANs are also quite different in that they are intended to pass traffic specifically for storage and storage management purposes only. In essence, a SAN is constructed from storage interfaces and has resulted in the creation of new methods of attaching storage to servers. As such, SANs have the potential to offer several major benefits to a JD Edwards-driven enterprise, such as:

- Improved performance
- Streamlined and centralized backup processes
- Facilitated sharing of data
- Centralized and consolidated storage
- Centralized storage management
- Host and storage attachment scalability
- Extended distance connectivity

A SAN is normally located near other computing resources but can also include high-speed connections with remote locations for disaster recovery-based backup, recovery, and for archival purposes. A SAN provides fibre channel connectivity between hosts (multiple, heterogeneous) and storage resources, for example disk subsystems and automated tape backup. A typical SAN environment is shown in Figure 8-1.

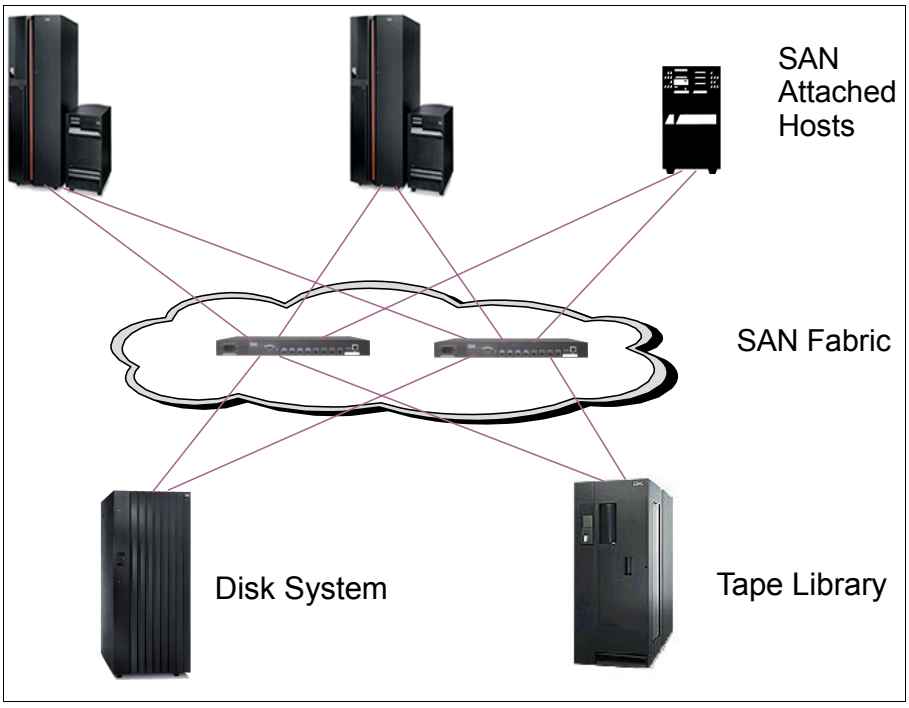

Figure 8-1 SAN overview

8.2 Overview of IBM SAN attached disk storage

IBM offers three external SAN disk options of IBM @server i5 and iSeries to meet a full range of performance, capacity, availability, and cost requirements:

- IBM TotalStorage DS6000 Family (1750)
- IBM TotalStorage DS8000 Family (2107)
- IBM TotalStorage Enterprise Storage Server® (2105)

This section describes each of these options to provide an understanding of the SAN disk solutions available.

8.2.1 IBM TotalStorage DS6000 series

The IBM TotalStorage DS6000 series is designed to deliver enterprise-class storage capabilities in a space-efficient, modular design at a low price. This functionality, as well as the high performance and advanced functions found in enterprise disk storage devices, is available in 19-inch rack mountable packages with the base storage server enclosure 5.25-inch (3U) high, and modular expansion enclosures. The IBM TotalStorage DS6800 (Model 511) and Expansion Enclosure (Model EX1) provide an intelligent disk system that delivers enterprise-class storage capabilities in a space-efficient 2Gbps fibre architecture.

The TotalStorage DS6800 consists of redundant, hot-swappable power supplies and cooling fan assemblies, redundant RAID Controllers, supporting RAID 5, RAID 10 and optional TotalStorage Resiliency Family functions, IBM TotalStorage FlashCopy®, Metro Mirror, Global Mirror, Metro/Global Copy. The DS6800 with Expansion Enclosures supports up to 224 disk drives for a total of up to 67.2 TB of storage. IBM TotalStorage DS6000 Expansion Enclosure attaches to the DS6800. 2Gbps Fibre Channel Disk Drive Sets drives are available in 73 GB 15,000 rpm, 146 GB 10,000 rpm, and 300 GB 10,000 rpm capacities.

8.2.2 TotalStorage DS8000 series

The TotalStorage DS8000 series is designed to provide unmatched functionality, flexibility, and performance for enterprise disk storage systems. It incorporates a high bandwidth internal fabric designed to support fault tolerance, highly expandable and flexible processor memory, and fibre channel attached disks. It has a dual processor complex implementation base on IBM POWER5 technology that supports concurrent microcode loads, transparent I/O failover and failback support, and redundant, hot-swappable components for 24 x 7 business environments. The maximum host I/O operations per second of a DS8300 is up to six times the maximum of an ESS Model 800. The DS8000 series also offer all new power and packaging which takes 20 percent less floor space than a base ESS Model 800. The DS8000 supports RAID-5 and RAID-10 disk protection. Both RAID-5 as well as RAID-10 protection are intermixable within a single system. The DS8000 models come standard with the IBM TotalStorage DS Storage Manager.

The IBM TotalStorage DS8000 Family is comprised of two series of models, the DS8100 and DS8300. The model DS8300 is offered as non-partitionable (model 922) and partitionable (model 92A) configurations. The DS8100 offers scalability of physical capacity from 1.1 to 115.2 TB. Fibre channel disks are available in 73 GB, 146 GB, and 300 GB capacities and these are intermixable within a single system. The DS8100 incorporates POWER5 processor technology from IBM in a dual 2-way processor-complex offering, the Model 921. The DS8300 offers scalability of physical capacity up to 192TB. Fibre channel disks are available in 73 GB, 146 GB, and 300 GB capacities, intermixable within a single system. The DS8300 incorporates POWER5 processor technology from IBM in two dual four-way processor complex offerings, the DS8300 Models 922 and 9A2. The DS8300 Model 9A2 enables the creation of multiple IBM TotalStorage Storage System logical partitions (LPARs).

8.2.3 IBM TotalStorage Enterprise Storage Server

This section describes the current models of the IBM TotalStorage Enterprise Storage Server (ESS).

ESS Model 800

The ESS Model 800 is designed to provide performance, scalability, and flexibility. It supports 24x7 operations to help provide the access and protection demanded by today's business environment and delivers the flexibility and centralized management needed to lower long-term costs. The Model 800 symmetrical multiprocessors (SMP) with an optional Turbo feature, 64 GB cache, double internal bandwidth, and 2 GB Fibre Channel/FICON Host Adapters. This hardware, in addition to RAID-10 support and 15 000 rpm drives, enables the Model 800 to deliver excellent levels of performance throughput. The ESS Model 800 has two clusters each contains RISC SMP processors. It supports 18.2, and 36.4, 72.8, and 145.6 GB DDMs and with an expansion enclosure the ESS Model 800 using 145.6 GB DDMs, can provide a maximum data storage capacity of 55.9 TB.

ESS Model 750

The ESS Model 750 is designed to help meet the needs of clients who do not require the full range of capacity and performance scalability offered by the ESS Model 800, but need enterprise level functionality intended to support reliable, continuous access to data at an affordable price. The ESS Model 750 provides an entry point into the ESS product family comprised of ESS Model 750 and ESS Model 800. The ESS Model 750 offers many of the features of ESS Model 800, but is based on a 2-way processor with 8 GB cache and 2 GB of Non Volatile Storage (NVS). With support for up to 4.6 TB of physical capacity and six 2 Gb Fibre Channel/FICON or ESCON® host adapters, the ESS Model 750 is designed to provide an attractive price/performance balance for clients requiring smaller configurations. The ESS Model 750 is well-suited for environments where the workload access densities average two operations per second per GB.

The IBM TotalStorage DS6000, DS8000, and ESS support critical requirements for strategic business initiatives, for example: on demand business, enterprise resource planning, business intelligence, service consolidation, and other mission-critical applications, all of which are a part of progressive JD Edwards customer sites. The IBM TotalStorage DS6000, DS8000, and ES support for TotalStorage Resiliency Family technologies from IBM. TotalStorage Resiliency Family from IBM includes FlashCopy, Global Mirror, and Metro Mirror.

The IBM TotalStorage DS6000, DS8000, and ESS are designed to support a broad range of operating environments including IBM @server zSeries, IBM @server i5 or iSeries, and pSeries servers, as well as servers from Sun™, Hp, and other Intel-based providers. For IBM @server i5 or iSeries attachment via 2Gbps Fibre Channel controllers are supported.

For additional product details and support information, see:

http://www-1.ibm.com/servers/storage/disk/index.html

8.3 IBM TotalStorage Resiliency Family for iSeries servers

The IBM TotalStorage DS6000, DS8000, and ESS offer several data mirror and replication solutions for the IBM @server i5 and iSeries platform. Referred to as the “IBM TotalStorage Resiliency Family” and “ESS Copy Services,” these disk copy solutions can be combined with iSeries availability solutions, including save-while-active (SWA) (nondisruptive backups), Domino® online backup, and iSeries clusters, to enjoy a broad range of options for meeting the commitment of the JD Edwards enterprise to its business availability objectives.
Figure 8-2 illustrates some of the technologies that make up the IBM TotalStorage Resiliency Family.

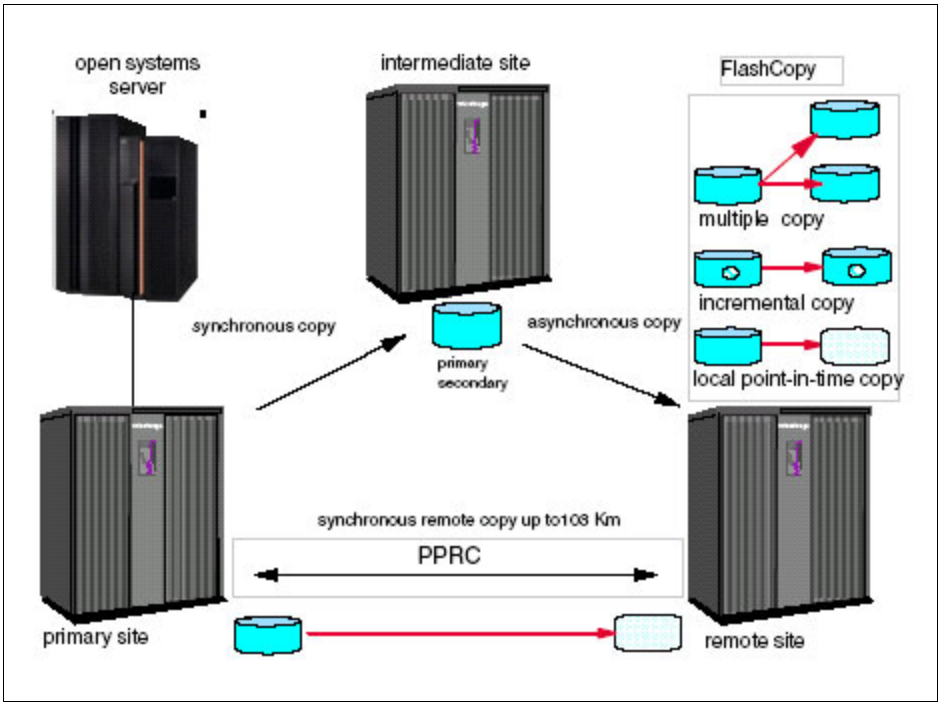

Figure 8-2 IBM TotalStorage Copy Services function overview

These disk copy solutions are not mutually exclusive, but they are not always interchangeable, either. Each solution has its own set of benefits and considerations. Let's look at these next, along with their relevance to the iSeries server.

8.3.1 IBM TotalStorage FlashCopy

IBM TotalStorage FlashCopy provides a near instantaneous copy of data. This helps minimize the downtime needed for backups. It creates a physical point-in-time copy of the data, making it possible to immediately access both the source and target copies. By creating an “instant” copy, FlashCopy enables applications using either the source or the target to operate with only minimal interruption to perform the FlashCopy.

Note: The use of Independent Disk Pools is required.

When FlashCopy is invoked, it creates a map of the source volumes. This process takes only a few seconds and when complete, you have access to the T0 copy of the source volumes. As soon as the map of the target volumes has been created, you can read and write to both the source and target volumes. The point-in-time copy created by FlashCopy is typically used where you need a copy of production data to be produced with minimal application downtime.

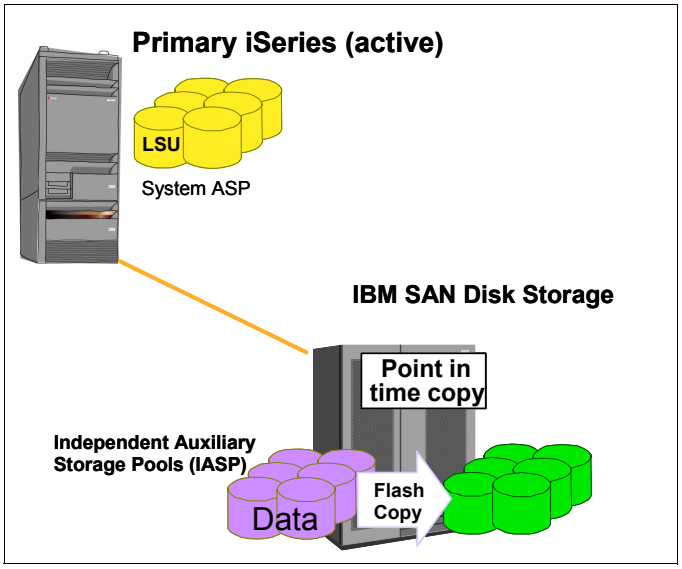

Figure 8-3 Flashcopy overview

Because of the nature of iSeries single-level storage, it is necessary to consider the Load Source Unit (LSU) as a special case. On other platforms, such as UNIX or Windows, each disk volume can be identified with its contents. The iSeries is different because all storage is considered as a single large address space. The LSU is within this address space. Therefore, mirror the LSU from the internal drive into the ESS to ensure the entire single-level storage is copied in order to use facilities such as PPRC or FlashCopy to perform a hardware copy of the disks attached to the iSeries.

iSeries independent disk pool, or independent auxiliary storage pool (IASP), is a collection of disk units that can be brought online or taken off-line independent of the rest of the disk storage on a system. ESS copy functions in combination with iSeries IASPs enable customers to use Flashcopy for taking backups with minimal downtime of production applications. This is only supported by using the iSeries Copy Services for ESS Toolkit offered by IBM Rochester Continuous Availability Team. The Toolkit provides necessary code and services for running ESS copy functions of an IASP.

You can use the copies of the data taken with FlashCopy to take offline backups or for other purposes, such as populating a DataWarehouse, providing a restart point for a long-running batch suite, or data migration. You can also use OS/400 mirroring in conjunction with the ESS. When using Flashcopy, you should plan for disk capacity requirements.

8.3.2 IBM Total Storage peer-to-peer-remote-copy

Peer-to-peer-remote-copy (PPRC) is a hardware-based disaster recovery solution that provides mirroring of logical volumes within an IBM TotalStorage DS6000, DS8000, and ESS or with another IBM TotalStorage DS6000, DS8000, and ESS. PPRC is an optional feature on these products. It is an established data mirroring technology that has been used for many years primarily to protect an organization's data against disk subsystem loss or complete site failure.

Note: The use of Independent Disk Pools is required.

When talking about PPRC, the systems where production applications run from are referred to as the primary or application site. Systems where the recovery or test of applications are

performed is referred to as the secondary or recovery site. PPRC is a synchronous or asynchronous protocol that allows mirroring of data from one Logical Unit (LUN) to another LUN. The LUNs can reside within the same IBM TotalStorage DS6000, DS8000, and ESS or in another located at a secondary site some distance away. PPRC is application independent. The copy functions are occurring at the disk subsystem level. You can use backup applications such as BRMS with PPRC and Flashcopy to provide offline, remote, backup, and recovery. Figure 8-4 provides a basic PPRC configuration.

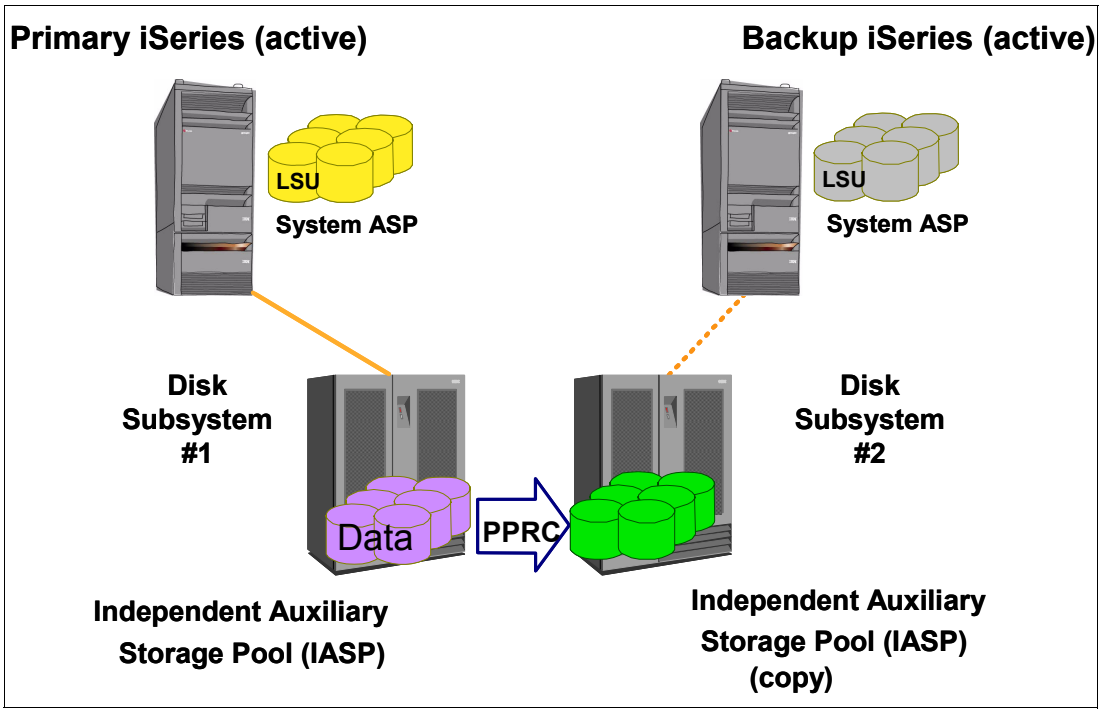

Figure 8-4 IBM PPRC overview

The IBM @server i5 and iSeries servers have specific requirements and have been tested to varying degrees with the PPRC modes. For further support details refer to *IBM @server iSeries in Storage Area Networks: A Guide to Implementing FC Disk and Tape with iSeries*, SG24-6220.

The IBM TotalStorage DS6000, DS8000, and ESS support the use of PPRC in the following modes:

- IBM Total Storage Metro Mirror (Synchronous PPRC)

  PPRC synchronous mode can be used for real-time data mirroring. In this mode, updates made on the primary ESS (local site) are synchronously shadowed to a secondary ESS (remote site). Since this is a synchronous operation, the distance between the primary and secondary ESS will affect the application response time. Therefore, when operating in this mode, PPRC has a standard maximum supported distances between the primary and secondary ESS:

  - ESCON PPRC Links: up to 103 km
  - Fibre Channel PPRC Links: up to 300 km

  Note that no intermix of PPRC link types is allowed within the same logical subsystem (LSS). Extending distances beyond 300 km is also possible. However, due to network configuration variability the customer must work with the channel extender vendor to determine the appropriate configuration to meet their performance requirements. IBM support approval for these longer distances can be requested by submitting a Request for

Price Quotation (RPQ). The RPQ should include information about distance between sites, the channel extension technology, and the type of telecommunications line, the amount of network bandwidth, the ESS capacity, and a general description of the workload.

- IBM TotalStorage Global Copy (PPRC Extended Distance)

  PPRC-XD is a asynchronous long distance copy option suitable for data migration and periodic offsite backup. With a non-synchronous operation, the distance between the primary and secondary ESS will have only a minimal effect on the application response time. Therefore, PPRC-XD can operate at very long distances.

- IBM TotalStorage Metro/Global Copy (three-site mirror)

  Asynchronous Cascading PPRC can be used to create three-site or two-site long-distance remote copy solutions. With synchronous PPRC, the PPRC secondary volume (involved in a PPRC synchronous relationship) can also simultaneously serve as a PPRC primary volume in a PPRC Extended Distance (PPRC-XD) relationship to the remote site.

- Global Mirror (Asynchronous PPRC)

  Asynchronous PPRC enables a high performance, asynchronous remote-mirroring copy solution. It is designed to provide a two site disaster recover and backup solution at virtually unlimited distances. Since the distance between the primary and secondary ESS has little impact to host applications at the primary site, the remote site can be located at distances from the local site well beyond those supported with Synchronous PPRC. The distances are typically limited only by the capabilities of the network and channel extension technologies.

An iSeries independent disk pool, or independent auxiliary storage pool (IASP), is a collection of disk units that can be brought online or taken off-line independently of the rest of the disk storage on a system. ESS copy functions in combination with iSeries IASPs enable customers to use Metro Mirror, Global Mirror and Global Copy with minimal recovery time. This is only supported by using the iSeries Copy Services for ESS Toolkit offered by IBM Rochester Continuous Availability Team. The Toolkit provides necessary code and services for running ESS copy functions of an IASP. As an exception to remote copy of IASP it is possible to use remote copy of entire iSeries disk space. This solution needs longer recovery time and does not prevent from a high availability exposure which is prevented by using Copy services of IASP.

8.4 IBM @server i5 and iSeries with IBM SAN Disk

Note: This information in this section is not intended to serve as a detailed sizing or implementation guide. For details about the iSeries server with SAN, see *IBM @server iSeries in Storage Area Networks: A Guide to Implementing FC Disk and Tape with iSeries*, SG24-6220.

The disk units used in IBM TotalStorage DS6000, DS8000, and ESS are referred to as disk drive modules (DDMs). These DDMs are packaged in groups of four or eight drives. In a DS6000, DS8000, and ESS, Logical Unit Numbers (LUNs) are the virtual representations of a disk as seen from the host system. LUNs are mapped over the DDMs. In reality, an LUN can span multiple physical disks. The size of the LUN is set when it is defined to the DS6000, DS8000, and ESS.

The iSeries server treats the SAN environment as a set of physical disk arms or spindles over which it has control. The SAN internally provides the iSeries server with LUNs or a set of

“virtual disks,” which for all practical purposes, behave exactly as a single disk to the iSeries server. The SAN manages the assignment of this LUN and its configuration apart from the iSeries server; it is transparent to the iSeries server. Once again, the LUN itself, configured by the SAN, can contain one or more disks and can be configured in a variety of RAID configurations. Optimization algorithms on the iSeries server treat this LUN as a single disk arm.

The introduction of a SAN requires additional system management resources, apart from that required to manage the iSeries server itself. This management is automatically performed by i5/OS or OS/400-only if the storage is internal. However, because the external storage is not completely under the control of the iSeries server, additional expertise and management are needed to maintain this resource. For example, the iSeries server does automatic performance tuning of a disk that has been made available from a SAN. In an attempt to increase performance, the SAN administrator makes changes to its configuration, such as increasing the size of the LUN. The iSeries server responds automatically to these changes and, in turn, makes system performance changes.

This adjusting continues until a steady state of the system is achieved. Since the result might not be the one as desired originally, considerable time and resources can be encountered in managing disk storage on the iSeries server. To avoid these problems, keep to a minimum the changes to the disk configurations. Care should also be taken to monitor changes and manage their effects.

Expert Cache in regard to the efficiencies it brings to internal storage on the iSeries platform is discussed in 8.7.2, “iSeries server expert cache” on page 113. It is important to note, though, that the Expert Cache technique (algorithm) for buffering data from the disk storage into memory in anticipation of its usage is also a key factor to improving the performance of the SAN storage. Expert Cache can perform much of the costly operations asynchronously to the application that is seeking data from a SAN network.

In at least one documented J.D. Edwards OneWorld (now JD Edwards EnterpriseOne) implementation with the ESS, both system throughput and response time performance increases were observed when Expert Cache was turned on. This indicates that the read/write caching algorithm of the ESS SAN, in general, works well with Expert Cache on the iSeries server to improve performance.

While IBM direct-attached storage remains a primary storage offering for iSeries, some JD Edwards customers have a requirement to deploy a SAN strategy. With the introduction of fibre channel connectivity to selected storage devices beginning with OS/400 V5R1, iSeries has significantly improved its SAN capabilities for both OS/400 and PowerPC® Linux running in a logical partition. OS/400 V5R2 builds on this strategy by further extending SAN participation via support for switch fabric topology and additional SAN network switches and directors.

8.5 Why iSeries and SAN

The IBM iSeries platform is an advanced, complex server capable of running many different types of workloads simultaneously.

Coupled with today's relentless onslaught of high-quality, must-have ancillary business applications, a diverse and varying demand has been created for the use of the same pool of storage. Native iSeries workloads such as ERP, online transaction processing, batch processing, UNIX, Linux, Lotus® Domino, Java and WebSphere, and Microsoft servers can be very I/O intensive and require a storage configuration designed to meet certain levels of performance. Other workloads, such as image management or file and print serving might

require less performance, but more capacity. Equally important, some workloads are cache friendly, while others have little benefit from large cache sizes. Gone are the days when designing a storage solution (on any platform) can be based solely on capacity requirements.

From the perspective of a pure iSeries server shop, an analysis must be made as to the reasons to implement SAN technology. The capabilities of the iSeries server internal drives and the single-level storage architecture provide some of the benefits of SAN technology including:

- Storage consolidation
- Storage management
- Disk reliability and protection

Disk capacity on the iSeries server is comparable to capacities from SAN vendors. Price performance on the iSeries server might be more cost effective compared to the SAN technology depending on the solution required. The true value of iSeries server and SAN technology interoperability is to provide customers the option of using SAN technology based on their business needs.

Implementations of SAN technology on an iSeries server as the primary platform are in many cases for the purpose of storage consolidation. Storage consolidation is the ability to connect many systems, not all of the same platform, to centralized disk storage. The connections can be direct or can include a hub or switch.

For additional information regarding the IBM SAN solutions, reference

http://www-1.ibm.com/servers/storage/san/

8.6 Performance effects between the iSeries and SAN

Performance situations can arise between the SAN and the iSeries server such as optimization conflicts between the disk cache algorithms on the iSeries server and SAN. Also, application program usage on the iSeries server can cause automatic changes in the disk cache allocation that is driven by the iSeries server disk cache algorithm. Alone, this is a very desirable effect because it enhances the overall server performance.

However, in the case of the SAN, another variable is introduced, namely the SAN disk cache. The SAN disk cache also attempts to adjust its allocation according to a different set of rules. The net effect is disk cache contention causing a less than optimal performance for both the SAN and iSeries server. Each system continues to adjust its cache automatically until a steady state is achieved. Performance events with iSeries server expert cache is discussed in 8.7.2, “iSeries server expert cache” on page 113.

8.6.1 Interoperability concerns

Interoperability concerns between the SAN and iSeries server include:

- Host Bus Adapter (HBA) cards

  HBA cards and their corresponding drivers are often overlooked as a possible candidate for problems. Consult an HBA card vendor Web site for any known situations with the iSeries server device drivers or connections to SAN devices. Most of the situations are with interoperability, and device driver patches for the iSeries server are readily available. SAN vendors also have an affinity to specific HBA card vendors.

  You should use a single HBA card vendor in all systems in a given SAN architecture.

- SAN switches, directors, and hubs

  SAN switches, directors and hubs and the software that controls them not only creates another level of management but introduces another point of failure and area of complexity for the JD Edwards EnterpriseOne / JD Edwards World software implementation. The general recommendation is to limit the number of switches and hubs if possible while trying to design for appropriate levels of SAN redundancy and availability

- Other iSeries server and non-iSeries server interaction

  Depending on the SAN configuration, there might be JD Edwards EnterpriseOne and JD Edwards World performance situations that are introduced with the sharing of centralized disk storage by iSeries server and other non-iSeries servers. iSeries servers use a different blocking size for disks than other systems, and therefore, iSeries servers' SAN LUN disks must be dedicated as part of the initial configuration of the SAN. These considerations are discussed in 8.7, “iSeries and SAN technology considerations” on page 113.

8.7 iSeries and SAN technology considerations

There are many factors to take into consideration when implementing SAN technology on an iSeries server. This section introduces some of the most common concerns.

8.7.1 Internal versus external storage

The iSeries server is designed and optimized on a distributed I/O processor (IOP) topology. This topology load balances the disk I/O and system processes equally in a parallel fashion. Internal and external storage is treated equally on disk I/O channels. Therefore, performance on the internal storage can be expected, by the nature of the shorter data path, to better scale and have higher throughput than the external SAN storage. The SAN data path connection, given the latency that is introduced into the data path by the distance and speed of the connection, creates overhead that can in complex SAN environments (high number of cascaded switches) contribute to decreased performance.

Performance on an iSeries server and SAN also depends on the processor, main memory, IOP processor, and SAN configuration. Carefully examine sources of response times and throughput to determine the primary bottlenecks. Increasing the speed and capability of one resource might bring only a short term or small benefit because another resource limitation is encountered. Use the iSeries server capacity planning tools to determine the current resource limitations at different conditions of use to decide whether there is a true issue of SAN performance. See Chapter 3, “JD Edwards EnterpriseOne 8.9 performance management” on page 25 for more information.

At low throughput, both internal and external disk subsystems perform at about the same level. The performance characterizations of iSeries server internal and external SAN disk storage can change dramatically at higher levels. Internal disk, in general, performs very well, but this depends on the configuration, setup, and application environment variables that contribute to overall performance.

8.7.2 iSeries server expert cache

Expert cache is a feature that buffers data from the disk storage into memory in anticipation of its usage. This key feature is important for increasing performance on applications that are read intensive. The expert cache algorithm is based both on the logical and physical characteristics of disk utilization. The expert cache mode can be selected for each memory

pool using the CHGSHRPOOL command with the paging options parameter of *FIXED, *CALC, or USRDFN as follows:

- When expert cache is OFF (*FIXED), there is no buffering of data.
- When expert cache is ON (*CALC), buffering of data occurs with the default strategy.
- When expert cache is ON (USRDFN), buffering of data occurs with the user defined strategy.

Expert cache improves the performance of both the internal and external SAN storage because it can perform much of the costly operations asynchronously to the application.

On at least one JD Edwards OneWorld and JD Edwards EnterpriseOne implementation with the IBM ESS, both system throughput and response time performance increases were observed when expert cache was turned on. This indicates that the read/write caching algorithm of the IBM ESS SAN, in general, works well with expert cache on the iSeries server to improve performance.

8.7.3 SAN storage cache

The read/write caching algorithm of an external SAN cache samples the data and adjusts the cache based on the current local sampling information without the benefit of viewing both the logical and physical resources.

SAN storage cache must be adequate enough to provide an extra read/write cache to mask resource conflicts that can occur as the iSeries server storage manager arm utilization parameters, adjusted with the SAN arm utilization parameters. Limited SAN storage cache can have unexpected results depending on the application and usage.

8.7.4 SAN configuration: Dedicated versus shared disk

Shared disk configuration refers to the layout of the storage on a SAN. In SAN implementations with iSeries servers, SAN disks are typically not shared with non-iSeries servers. The scope of this discussion revolves around the sharing of a set of SAN disk assignments between one or more iSeries servers connecting to the same SAN.

A dedicated SAN configuration reserves an entire set of disks (called a disk array, disk group, or rank) to be used exclusively by a single iSeries server. In this configuration, the SAN subsystem partitions its internal disk, and in some cases its cache, between a single iSeries server and other systems. The advantage of this configuration is that there is no contention for these resources with other systems. The single iSeries server operating system does not struggle to optimize the performance of the LUNs that the SAN presents to the system as virtual drives.

For the shared environment, there are two main configurations:

- In a two-system iSeries server configuration, two separate disk arrays are configured. Each array is assigned to a specific iSeries server.
- A single disk array is created to serve both iSeries servers. The LUNs created from this single disk array are split between these two systems.

The performance on a dedicated subsystem, in general, outperforms a shared configuration. The reason is that both the iSeries server and SAN resources depend on resources such as cache, disk arms, processors, memory, and buses. When more than one request for these resources is needed, the system must split those resources among the requesting agents. Such is the case of a shared configuration. In the case of a dedicated system configuration, there is a lower conflicting overhead and, therefore, an anticipated increase in performance.

Disk operations on a SAN do not provide a mechanism for request prioritization based on application or type of request. Therefore, a system with a large disk storage volume using a SAN can be expected to perform at a lesser level than that of internal storage.

8.7.5 Multi-pathing from iSeries to SAN

i5/OS V5R3 has a new built-in capability for multi-pathing. If there is more than one path to the same set of ESS LUNs, i5/OS V5R3 will establish multipathing for them automatically. There can be up to eight paths for the same set of LUN.

Multiple paths to the same set of LUNs by assigning them to multiple iSeries FC adapters. After we do this, iSeries recognizes the multiple paths to the same set of LUNs and starts to use them: It spreads the I/O operations to these LUNs across all available paths using “Round Robin” rule to determine through which path will an I/O operation go. If one path will become unavailable because of failure of iSeries FC adapter, iSeries bus, ESS Host adapter, and so forth, the I/O load for the same LUNs will be transferred automatically to other available paths. Multi-path with OS/400 V5R3 has the following characteristics:

- Support for two to eight paths for each LUN
- No single failure on any path will result in loss of access to disk units (LUNs)
- Either point to point of via multiple switches or directors
- 32 LUNs per fibre maximum

The ability to have multiple redundant paths to a single LUN (multi-path), either direct attached or through a redundant SAN, is illustrated in Figure 8-5.

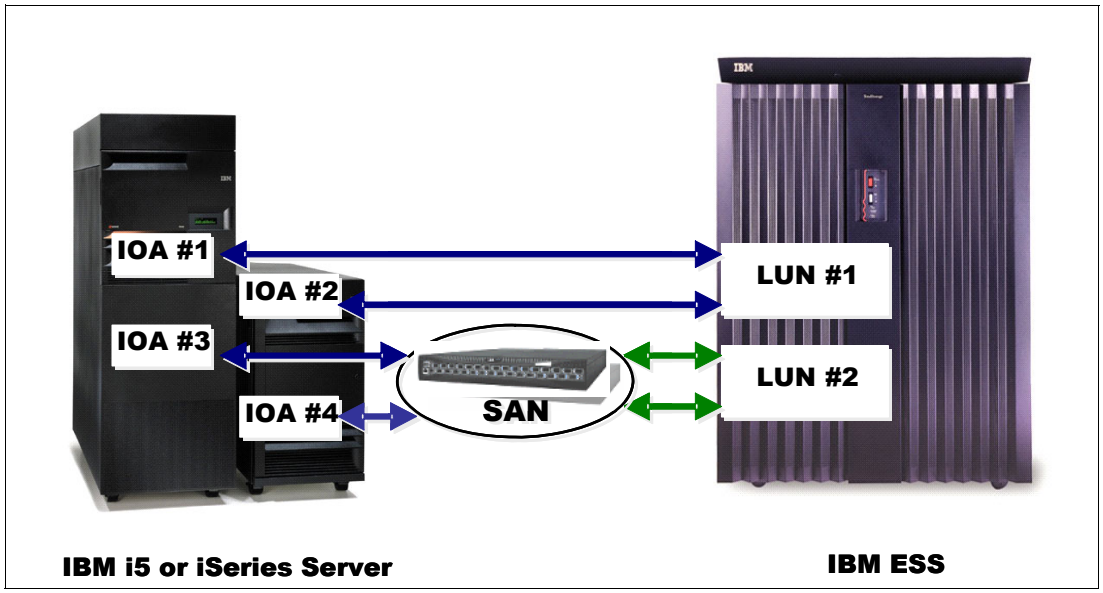

Figure 8-5 IBM iSeries i5/OS V5R3 Multi-Path support

Note: This redbook is written to an OS/400 V5R2 level. The i5/OS V5R3 content is left in this chapter to illustrate some of the advantages in the latest operating system for the iSeries servers.

8.7.6 Connectivity of iSeries to SAN

On SAN implementations with iSeries servers, a fibre connection directly to the disk subsystem can be made. Since the iSeries server treats internal and external disks the same,

a direct connection to the SAN from the iSeries eliminates any initial configuration issues dealing with the complexity of switch and hub architecture and the software needed to manage them. Large numbers of additional SAN hops, like those encountered through a several cascaded hubs, switches or directors, can affect the overall latency and application response times. Care must be taken to limit the introduction of base performance inhibitors such as cascaded SAN switches, hubs, and extended distance connections.

8.8 JD Edwards EnterpriseOne and JD Edwards World with SAN advantages

JD Edwards EnterpriseOne and JD Edwards World solutions have evolved into an equally advanced and complex set of mission-critical software packages that drive the core functions of literally thousands of enterprises around the world. In addition, the ability to connect UNIX, Windows NT®, and iSeries servers to the same storage has been the major factor in all of the JD Edwards EnterpriseOne or JD Edwards World SAN implementations.

The capabilities of iSeries internal drives, coupled with single-level storage, deliver a powerful argument for “resting content” with a sophisticated, cost-effective, effortless, internal storage solution. However, it is important to note that for a growing number of JD Edwards customers, the value of the interoperability provided between iSeries servers and SAN technologies is that it provides greater storage choices, based on the needs of the enterprise. The requirement for this expanded storage flexibility is evidenced by the fact that SAN implementations are becoming more common in JD Edwards organizations.

What are these business needs? The solutions a SAN brings to a JD Edwards EnterpriseOne or JD Edwards World customer, over and above the de facto benefits offered by the internal storage that is integrated into the i5 and iSeries platforms, include:

- Shared disk subsystem and tape device among multiple hosts

  By using Fibre Channel support, it is possible to attach more hosts to a single tape device via hubs, switches or directors. Disk storage subsystems can attach to multiple and heterogeneous hosts to save management costs and provide better resource use.

- DASD and tape device extended distance

  One of the major drawbacks of SCSI attached devices is the attachment limit of 25 meters (27 feet). Fibre Channel connection allows disk storage systems (DS6000, DS8000, ESS) and tape devices (IBM Linear Tape Open (LTO), IBM TotalStorage 3592 Tape Drive) to be located up to 300 meters from a single iSeries server (also depends on the fibre being used and the data transfer speed). Using cascaded switches and long wave fibre channel technology, devices can be located over 10 kilometers from the server. Distances greater than 10 kilometers are also possible using distance extension solutions such as DWDM or WAN based channel extenders.

- Disaster recovery

  iSeries-based cluster solutions provide high availability by addressing all outage types, including server failures, server and application maintenance and upgrades, and disaster recovery. Recovery time from unplanned outages is usually as little as a few minutes or as much as one or two hours, with little or no data loss. Options such as IBM TotalStorage FlashCopy (Point in Time Copy), IBM TotalStorage PPRC (Metro Mirror, Global/Remote Copy) can significantly reduce downtime.

Note: You can use both IBM external SAN disk subsystems and iSeries Integrated Storage with iSeries Clusters

- Time to save

  Many JD Edwards customers have requirements to reduce or eliminate their nightly backup windows. In an iSeries environment there are three technologies that can address these needs: IBM TotalStorage FlashCopy (Point in Time Copy), IBM TotalStorage PPRC (Metro Mirror, Global/Remote Copy), OS/400 Save-While Active (SWA), and iSeries clusters. Each of these solutions offers different benefits. Some of these solutions require downtime on the production (primary) i5/iSeries server, which can include ending applications, then restarting them. FlashCopy is discussed in 8.3.1, “IBM TotalStorage FlashCopy” on page 107.

- Storage consolidation

  SAN technology is often used to consolidate and standardize the management of disparate storage sources. Storage consolidation is the ability to connect many systems and many platforms, into a centralized disk storage representation. The connections can be direct or can include a fibre channel hub, switch, or director. The ability to connect UNIX, Windows, and iSeries servers to the same storage is a major factor in all of the JD Edwards SAN implementations to-date.

- Tape device performance

  Fibre Channel technology provides the potential for faster data rates than earlier tape connection technologies. The actual performance realized, however, depends on the application and data type used. Other factors do influence SAN performance, for example, the number of host servers and tape devices attached to a switch; zoning; and if switches are cascaded, the number of Inter Switch Links (ISLs).

  Note: Zoning refers to a basic technique for SAN management that can be implemented in one of many methods-all of which are intended to restrict access by a user to unauthorized storage devices. Zoning serves to heighten security and decrease unnecessary traffic over the network.

8.9 JD Edwards EnterpriseOne and JD Edwards World implementation considerations

There are a number considerations that can minimize the potential for issues that are related to interoperability between the iSeries server and the SAN. You should contact the SAN vendor for information concerning recent issues, updates, and bug fixes with the iSeries server. SAN vendors maintain a list of supported Host Bus Adapter cards, and you should address issues that concern connectivity to the SAN vendor.

8.10 IBM TotalStorage SAN Disk with JD Edwards

The IBM TotalStorage DS6000, DS8000, and ESS are an attractive SAN choice for JD Edwards users. They support the enterprise goal of comprehensive enterprise disk storage consolidation. They allow for data sharing on multiple, heterogeneous servers, with combined platform storage capability. For example, the DS6000 model scales to 67.2 TB, and the DS8000 scales to 192 TB and the ESS Model 800 scales to 55.9 TB. They also provides high-performance RAID-5 and RAID 10 data protection options, read and write cache, and the flexibility of common storage for multiple servers with differing channel attachments.

The IBM TotalStorage DS6000, DS8000, and ESS bring advanced storage capabilities to the iSeries platform by allowing more storage consolidation and flexibility in an enterprise

environment, such as multi-server connectivity and increased storage allocation flexibility. In many client environments, the strategic direction is that all servers utilize SAN-based storage. The IBM TotalStorage DS6000, DS8000, and ESS are designed to meet those needs excellently, and allows JD Edwards clients to continue enjoying a one-vendor solution for their server needs.

When the IBM TotalStorage DS6000, DS8000, and ESS are attached to an iSeries server, they becomes an inherent extension of iSeries storage and of the iSeries automated storage management capabilities. Thus, the IBM TotalStorage DS6000, DS8000, and ESS also benefit from the iSeries automated data distribution, graphical storage management, availability capabilities, integrated backups, and the ability to support every type of iSeries workload, all the while requiring essentially no greater user attention to storage management than is needed when only internal iSeries storage is utilized.

JD Edwards clients observe that IBM SAN disk subsystems provide a storage solution with flexible and superior storage consolidation.

8.11 Summary

SAN technology brings benefit in an e-commerce application such as JD Edwards EnterpriseOne or JD Edwards World. The distributed nature of the different component systems that make up an ERP implementation can be extensive. SAN technology can be used to simplify the data storage and maintenance of at least some of these components.

The IBM TotalStorage DS6000, DS8000, and ESS are state-of-the-art solutions from IBM for accomplishing a SAN implementation with each JD Edwards application set. The IBM TotalStorage DS6000, DS8000, and ESS support data from a variety operating system platforms. It allows all the different data storage pockets of information to be consolidated into a single data server environment while still making them completely available, without any encumbrances to the various users who need to get to them.

By its very nature, a chapter such as this cannot be specific in the information and guidance it gives to individual JD Edwards clients. It is the intent of both Oracle and the IBM @server iSeries server group to answer the unique concerns regarding this SAN solution. Contact an IBM representative or IBM Business Partner to gain a much more customized perspective on these new technologies that are rapidly gaining popularity.

A

The JD Edwards EnterpriseOne 8.9 test environment and transaction details

This appendix provides detailed background information for the tests that were performed with JD Edwards EnterpriseOne 8.9 Web server and the Virtualization benchmark.

© Copyright IBM Corp. 2005. All rights reserved.

JD Edwards EnterpriseOne 8.9 Web server test environment

This section describes the test environment referenced in Chapter 3, “JD Edwards EnterpriseOne 8.9 performance management” on page 25.

Tests of the JD Edwards EnterpriseOne 8.9 Web client solution were conducted with both an All-In-One (AIO) and a Virtual Three Tier (V3T) environment. All tests were performed on iSeries servers running OS/400 V5R2, EnterpriseOne Release 8.9 (or higher), WebSphere 5.0, and HTTP Apache server. Figure A-1 shows the V3T environment and the components that ran on each system.

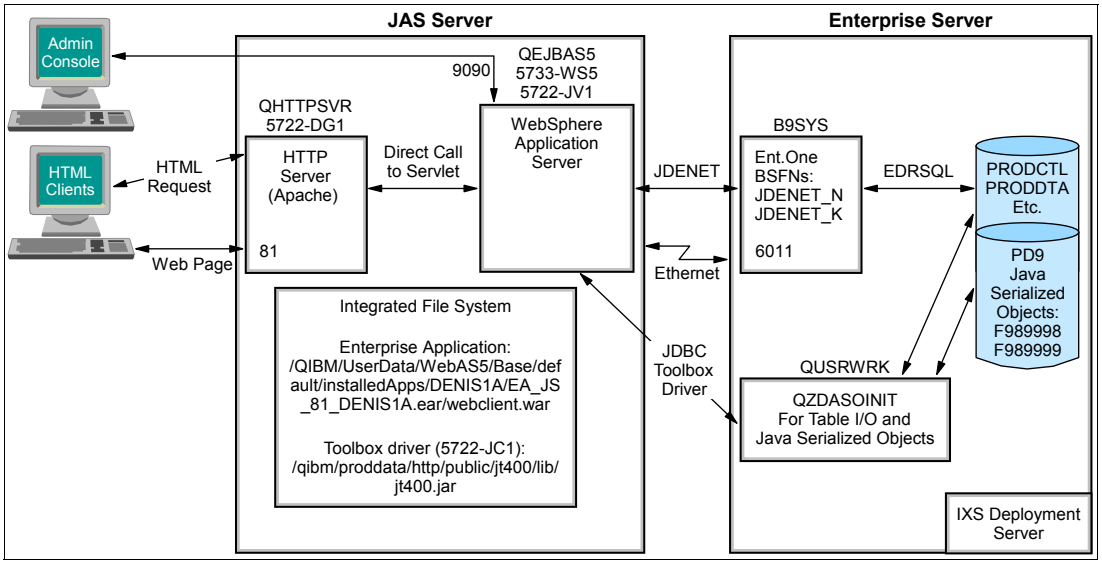

Figure A-1 Virtual Three Tier (V3T) configuration

Table A-1 shows the hardware that was used during the tests to make the recommendations in the Chapter 3, “JD Edwards EnterpriseOne 8.9 performance management” on page 25. Tests were run on an iSeries Model 825 #2473 processor partitioned into a 3-way and a 2-way system. Table A-1 describes the configuration of the enterprise server partition.

Table A-1 iSeries hardware testing environment (enterprise server)

Hardware	Description	Number or Amount
Processors	3600 CPW	3
Disk drives, RAID	15 arms (35 GB drives)	457 GB
Memory		14 GB
I/O card	1 Gb Ethernet	1

The configuration partition of the Java server system is illustrated in Table A-2.

Table A-2 iSeries hardware testing environment (Web server)

Hardware	Description	Number or Amount
Processors	2400 CPW	2
Disk drives, RAID	15 arms (35 GB drives)	457 GB
Memory		8 GB
I/O card	1 Gb Ethernet	1

Table A-3 shows the software and group PTF levels in use in the test environment to make the recommendations in Chapter 3, “JD Edwards EnterpriseOne 8.9 performance management” on page 25.

Table A-3 iSeries software testing environment

Software (WRKPTFGRP)	Description	Service Pack
OS/400	V5R2	Cum PTF C4077520
Database	SF99502	Service Pack #12
EnterpriseOne	8.9	SP2_B1
HTTP	SF99098	Service Pack #13
WebSphere Advanced Base 5.0.2	SF99245	Service Pack #7
Java	SF99169	Service Pack #16

It is highly recommended that the production environment be kept current with the JD Edwards informational APAR.

The following link is for the technical resources for the JD Edwards EnterpriseOne product and contains links to the informational APARs for OS/400 V5R2 and i5/OS V5R3:

http://www.ibm.com/servers/enable/site/events/jde_links.html

Virtualization benchmark JD Edwards Enterprise One 8.9 transaction details

This section provides further information for the virtualization benchmark described in Chapter 6, “Virtualization for server and application consolidation” on page 71. The transaction details involved in the The JD Edwards EnterpriseOne 8.9 and JD Edwards EnterpriseOne Learning Management 8.81 applications are described.

Table A-4 shows the transaction details and workload mix for the JD Edwards Enterprise One 8.9 workload.

Table A-4 EnterpriseOne 8.9 transaction details and workload mix.

User group	Transaction	Documents	Run	Lines—all documents
Manufacturing	Work order entry*	1 document, 1 line	4	4
	Work order completion*	1 document, 1 line	4	4
	Other manufacturing documents	3 documents	12	12
	Total manufacturing transactions	5	20	20
Distribution	Purchase order entry*	1 document, 10 lines	4	40
	Sales order entry*	1 document, 10 lines	4	40
	Other distribution documents	3 documents	12	12
	Total distribution transactions	5	20	92
Financials	Voucher entry*	1 document, 5 lines	6	30
	Journal entry*	1 document, 20 lines	4	80
	Other financial documents	6 documents	14	14
	Total financial transactions	8	24	124
Total Business Transactions Run		64		
Total Line Items			236	

Note: A weighted average transaction response time was collected across major transactions.

JD Edwards Enterprise Learning Management 8.81 transaction details

Table 8-1 shows a detailed breakdown of the JD Edwards Enterprise Learning Management 8.81 transactions.

Table 8-1 Enterprise Learning Management 8.81 transactions

Transaction group	Benchmark percentage	Transaction description	Transaction pace (transactions per minute)
Learner Self Service	32%	Browse course catalog	6-8
	8%	Search catalog	6-8
	8%	Add learning to plan from catalog	6-8
	3%	Enroll in blended activity	4-6
	15%	Launch Web-based content	4-6
	1%	Register in program	8-10
Manager Self Service	8%	Approve learning	6-8
	4%	Add to learner's plan	8-10
	2%	Enroll team member	8-10
	8%	Add team member's objective	8-10
	4%	Review team learning history	4-6
Back Office / Call Center	3%	Process enrollment request	12-18
Instructor	4$	Mark grades and attendance	15-20

B

JD Edwards EnterpriseOne 8.9 test configuration files

This appendix lists the configuration files for HTTP and the EnterpriseOne Java and enterprise servers. The environment is configured for 200 users.

Note: The WebSphere configuration settings are edited through the Administrative Console. Therefore, a listing is not included in this appendix.

© Copyright IBM Corp. 2005. All rights reserved.

HTTPD.CONF file

Example B-1 lists the contents of the HTTPD.CONF file.

Example: B-1 The HTTPD.CONF file

```
HTTPD.CONF
LoadModule deflate_module /QSYS.LIB/QHTTPSVR.LIB/QZSRCORE.SRVPGM
WebSpherePluginConfig /QIBM/UserData/WebAS5/Base/default/config/cells/plugin-cfg.xml
LoadModule ibm_app_server_http_module /QSYS.LIB/QEJBAS5.LIB/QSVTIHSAH.SRVPGM
# Configuration originally created by Create HTTP Server wizard on Thu Jan 29 23:53:42 UTC
Alias /jde
/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1B/EA_JS_81_DENIS1B.ear/webclient.war
Listen *:81
DocumentRoot /www/vh_js_81/htdocs
Options -ExecCGI -FollowSymLinks -SymLinksIfOwnerMatch -Includes -IncludesNoExec -Indexes -
MultiViews
#LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
#LogFormat "%{Cookie}n \"%r\" %t" cookie
#LogFormat "%{User-agent}i" agent
#LogFormat "%{Referer}i -> %U" referer
#LogFormat "%h %l %u %t \"%r\" %>s %b" common
#CustomLog logs/access_log combined
LogMaint logs/access_log 7 0
LogMaint logs/error_log 7 0
LogLevel Error
MaxKeepAliveRequests 0
KeepAlive On
SetOutputFilter DEFLATE
SetEnvIf "User-Agent" "Mozilla/2" nokeepalive
SetEnvIf "User-Agent" "JDK/1\.0" force-response-1.0
SetEnvIf "User-Agent" "Java/1\.0" force-response-1.0
SetEnvIf "User-Agent" "RealPlayer 4\.0" force-response-1.0
SetEnvIf "User-Agent" "MSIE 4\.0b2;" nokeepalive
SetEnvIf "User-Agent" "MSIE 4\.0b2;" force-response-1.0
<Directory />
   Order Deny,Allow
   Deny From all
</Directory>
<Directory /www/vh_js_81/htdocs>
   Order Allow,Deny
   Allow From all
</Directory>
<Directory
/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1B/EA_JS_81_DENIS1B.ear/webclient.war
>
   Order Deny,Allow
   Allow From all
   AddOutputFilterByType DEFLATE text/html
</Directory>
<Directory
/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1B/EA_JS_81_DENIS1B.ear/webclient.war
/WEB-INF>
   Order Deny,Allow
   Deny From all
</Directory>
```

The jas.ini file

Example B-2 lists the contents of the JAS.INI file.

Example: B-2 The jas.ini file

```
JAS.INI
# Please refer to the JAS Installation Guide for detailed information on this file.
#
[SECURITY]
DataSource=System - B9
SecurityServer=DENIS1A
UseLogonCookie=FALSE
CookieLifeTime=7
SSOEnabled=FALSE
SSOUserIDOnly=FALSE
# set password only if SSOUserIDOnly is set to TRUE
Password=

[OWWEB]
PathCodes=('PD9')
MO
QUEUE=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webclie
nt.war/moqueue
MAXUser=450
UseMOWinNTShare=FALSE
SystemDateFormat=MDE
SystemDateSeparator=/

# The HelpPath key determines where JAS looks for help files.
# Syntax: "http://[machine]/[path]/".
# Note: if you specify "/jde/owhelp/", this machine is assumed.
HelpPath=/jde/owhelp/
OWJRNL=OWJRNL
# Library Name for the AS/400 STRJOURNAL stored procedure

[CACHE]
# These intervals are in milliseconds. For example, 60000ms = 1 minute
UserSession=2400000
CacheCheck=60000

[SERVER COMPONENTS]
#
# Do not modify this section
#
com.jdedwards.jas.UserManager
com.jdedwards.jas.JDBCProxy
com.jdedwards.jas.JDEORB
com.jdedwards.jas.DDValidation
com.jdedwards.jas.security.SecurityBroker
com.jdedwards.jas.UDCJDBC
com.jdedwards.jas.JDEUDCText
com.jdedwards.jas.JDEUpdates
com.jdedwards.jas.JDEQueries
com.jdedwards.jas.JDEOWDirect
com.jdedwards.jas.ServerQuery
com.jdedwards.jas.JDESignon
com.jdedwards.runtime.virtual.MAFComponentLoader

[JDENET]
```

```
# Enterprise Server JDENET port
serviceNameConnect=6011

# JAS Server JDENET Listening port
serviceNameListen=6010

# Maximum number of dedicated kernel types
maxKernelRanges=1
maxPoolSize=450
# Time out value for requests sent to the Enterprise Server
enterpriseServerTimeout=90000

# Directory for JDENET temporary file
tempFileDir=/tmp/InstallShield

# Clustered logical datasource retry interval
ClusterRetryInterval=60000

# Dedicated Kernel Definitions
[JDENET_KERNEL_DEF1]
krnlName=XAPI Kernel
processClassName=com.jdedwards.runtime.virtual.xapi.PublishedMessageProcessor
startMessageRange=15001
endMessageRange=15250
maxNumberOfThread=3


[SERVER]
# Enterprise Server and port providing Glossary Text information
glossaryTextServer=DENIS1A:6011

# Encoding scheme
# 1252- English and Western European
# 932- Japanese
# 950- Traditional Chinese
# 936- Simplified Chinese
# 949- Korean
codePage=1252

[LOGS]
# The STDERR and STDOUT keys must point to the same directory that was supplied for the
# STDERR and STDOUT fields when defining the WebSphere Application Server.
# This is usually the INTERNET directory under B7333.
Debug=FALSE
rtdebugTrace=FALSE
jdbcTrace=FALSE
#rtlogType=ALL
stderr=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webcli
ent.war/logs/stderr.log
stdout=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webcli
ent.war/logs/stdout.log
rtlog=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webclie
nt.war/logs/rt.log
#rtdebug=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webc
lient.war/logs/rtdebug.log

[JAS INSTANCE]
# This is for JAS redirector only
# RoundRobin will be used as default if ther is no entry in [JAS WEIGHT].
# If redirect based on JAS instance weight, then each instance's weight must specified,
```

```
# otherwise, weight 0 will be assumed, which means the server will not get
# any redirected request
# example: JAS1=http://127.0.0.1/jde
# Note: in this example, "JAS1" is the key, if weight will be given,
# this key must be used in [JAS WEIGHT] section to match this JAS instance

[JAS WEIGHT]
# The key must match the key in [JAS INSTANCE] section
# example: JAS1=2
# The weight is ratio based, for example:
# JAS1=2
# JAS2=1
# That means JAS1 will take as twice load as JAS2


[REDIRECTOR]
# interval unit is milliseconds
Interval=5000
# This is the location that Redirector will redirect to, if no server is available.
# If it is empty, redirect will display a simple page tells user that no server is
available
# example: NoServerAvailableURL=http://servername/path/noserver.html
NoServerAvailableURL=

[TRANSACTIONS]
# Configuration settings to customize transaction processing within JAS.
# ReapInterval      : Interval in minutes of how often the TransactionReaper
#                     should check for abandoned transactions.
#                     Valid values:  Integer from 1 to n
#                     Default: 5 minutes
# MaxAge            : Maximum age in minutes of a transaction.  Transactions
#                     older than this will be rolled back by the
#                     TransactionReaper.
#                     Valid values:  Integer from 1 to n
#                     Default: 10 minutes
#

ReapInterval=5
MaxAge=10

[ERPINTERACTIVITY]
#interactivity level. valid values are: LOW, MEDIUM, and HIGH
InteractivityLevel=HIGH
# whether multiple browsers fetch is enabled (turned off in lab to match scripts for
testing)
MultipleBrowserEnabled=FALSE
# the maximum number of open browsers for each user
MaxOpenBrowsers=10
# the number of records fetched for each database fetch before getting a system warning
DBFetchLimitBeforeWarning=2000


jdelog.properties
# This is a sample configuration file for root configuration
jdelog.rootLogger=ERROR,JDELOG
#jdelog.rootLogger=DEBUG,JDELOG,JASLOG
jdelog.loggerFactory=com.jdedwards.base.logging.log4j.JdeLoggerFactory
jdelog.reloadInterval=60

# These are the settings for file handlers.
```

```
jdelog.handler.JDELOG=com.jdedwards.base.logging.log4j.FileHandler
#jdelog.handler.JDELOG.File=jderoot.log
jdelog.handler.JDELOG.File=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_8
1_DENIS1A.ear/webclient.war/logs/jderoot.log
jdelog.handler.JDELOG.Level=ERROR
jdelog.handler.JDELOG.Append=TRUE
jdelog.handler.JDELOG.MaxBackupIndex=1
jdelog.handler.JDELOG.MaxFileSize=10MB
jdelog.handler.JDELOG.format=com.jdedwards.base.logging.log4j.DefaultFormat

# These are the settings for file handlers.
jdelog.handler.JASLOG=com.jdedwards.base.logging.log4j.FileHandler
#jdelog.handler.JASLOG.File=jas.log
jdelog.handler.JASLOG.File=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_8
1_DENIS1A.ear/webclient.war/logs/jas.log
jdelog.handler.JASLOG.Level=ERROR
jdelog.handler.JASLOG.Append=TRUE
jdelog.handler.JASLOG.MaxBackupIndex=1
jdelog.handler.JASLOG.MaxFileSize=10MB
jdelog.handler.JASLOG.format=com.jdedwards.base.logging.log4j.DefaultFormat

#This is added for debug logging. Please uncomment following to start debug.
#jdelog.Debug=DEBUG, jasdebug
#jdelog.handler.jasdebug=com.jdedwards.base.logging.log4j.FileHandler
##jdelog.handler.JASDEBUG.File=jasdebug.log
#jdelog.handler.jasdebug.MaxBackupIndex=5
#jdelog.handler.jasdebug.MaxFileSize=10MB
#jdelog.handler.jasdebug.File=/qibm/userdata/webas5/base/default/installedApps/DENIS1A/EA_J
S_81_DENIS1A.ear/webclient.war/logs/jasdebug.log
#jdelog.handler.jasdebug.Level=DEBUG
```

The JDBj.INI file

Example B-3 lists the contents of the JDBj.INI file.

Example: B-3 The JDBj.INI file

```
#============================================================================
#                    jdbj.ini - JDBj configuration
#
# This file contains configuration information for JDBj, which provides
# general database access capabilities on behalf of J.D. Edwards Java
# solutions.
#============================================================================

#============================================================================
#                    OneWorld specific information
#
# These sections apply only when accessing OneWorld data.  These
# sections are ignored for standalone database access.
#============================================================================

#----------------------------------------------------------------------------
# Bootstrap session
#
# This information is used to sign on to provide access to system
# tables.  JDBj signs this user on via the security server at bootstrap-
```

```
# time, so if it is not valid, then no OneWorld database access is
# possible.
#
# NOTES: * This information in this section should match the
#          corresponding information specified in the
#          [DB SYSTEM SETTINGS] section of this installation's JDE.INI
#          file.
#--------------------------------------------------------------------

[JDBj-BOOTSTRAP SESSION]
user=JDE
password=JDE
environment=PD9
role=*ALL

#--------------------------------------------------------------------
# Bootstrap physical data source
#
# This defines the data source where the OCM and some other system
# tables reside.  JDBj uses this at bootstrap time and later to look up
# OCM entries on demand.  If it is not valid, then no OneWorld database
# access is possible.
#
# Here is a description of each field in this section:
#
# ---------------  ---------------------------------------  -----------
# Field            Description                              Applies to
# ---------------  ---------------------------------------  -----------
# name             The data source name.  This is not that  All
#                  important for bootstrap connections,
#                  but it does show up in error messages
#                  and the log.
# ---------------  ---------------------------------------  -----------
# databaseType     The database type.  One of the           All
#                  following:
#
#                  I = AS/400
#                  O = Oracle
#                  S = SQL Server
#                  W = UDB
# ---------------  ---------------------------------------  -----------
# server           The server (i.e. the host).              AS/400,
#                                                           SQL Server
# ---------------  ---------------------------------------  -----------
# database         The database.                            Oracle, UDB
# ---------------  ---------------------------------------  -----------
# serverPort       The server port.                         SQL Server
# ---------------  ---------------------------------------  -----------
# physicalDatabase The physical database (used as library   AS/400,
#                  qualifier for AS/400).                   SQL Server
# ---------------  ---------------------------------------  -----------
# owner            The owner.                               Oracle,
#                                                           SQL Server,
#                                                           UDB
# ---------------  ---------------------------------------  -----------
# lob              Supports lobs.                           Oracle,
#
#                                                           AS/400
# ---------------  ---------------------------------------  -----------
# unicode          Performs unicode conversion.             SQL Server
```

```
# ---------------  -------------------------------------  -----------
#
# NOTES: * This information in this section should match the
#          corresponding information specified in the
#          [DB SYSTEM SETTINGS] section of this installation's JDE.INI
#          file.
#--------------------------------------------------------------------

[JDBj-BOOTSTRAP DATA SOURCE]
name=System - B9
databaseType=I
server=DENIS1A
serverPort=
database=
physicalDatabase=SY9
owner=
lob=true
unicode=true
#--------------------------------------------------------------------
# Bootstrap spec data source (Optional)
#
# This defines the data source where the serialized spec tables reside.
# JDBj uses this at bootstrap time and after to look up serialized spec
# objects on demand.  If this section is not specified, then JDBj will
# use the OCM to find the serialized spec table.  If it is not valid, no
# OneWorld database access is possible.
#
# See the description for "Bootstrap physical data source" above for
# information about which fields must be filled in for a given database
# platform.
#--------------------------------------------------------------------

[JDBj-SPEC DATA SOURCE]
name=Central Objects - PD9
databaseType=I
server=DENIS1A
;serverPort=
;database=
physicalDatabase=COPD9
;owner=
;lob=false
lob=true
;lob and unicode changed to true 3/8/04, why doesn't aio need jde/jde like V3T?
unicode=true
;user=
;password=

#====================================================================
#                END - OneWorld specific information
#
# The rest of the sections apply when accessing OneWorld and standalone
# data.
#====================================================================

#--------------------------------------------------------------------
# JDBC drivers
#
# This defines JDBC drivers and JDBC specific settings.
#--------------------------------------------------------------------
```

```
[JDBj-JDBC DRIVERS]
;ORACLE=oracle.jdbc.driver.OracleDriver
AS400=com.ibm.as400.access.AS400JDBCDriver
;SQLSERVER=com.microsoft.jdbc.sqlserver.SQLServerDriver
;UDB=COM.ibm.db2.jdbc.app.DB2Driver

[JDBj-ORACLE]
tns=/QIBM/UserData/WebAS5/Base/default/installedApps/DENIS1A/EA_JS_81_DENIS1A.ear/webclient
.war/WEB-INF

[JDBj-LOGS]
jdbcTrace=false

[JDBj-SERVER]
dbcsConversionTolerant=true

#-------------------------------------------------------------------------
# Connection pool settings
#
# These are used only when J2EE connection pooling is not available.
#-------------------------------------------------------------------------

[JDBj-CONNECTION POOL]
jdbcPooling=false
minConnection=5
maxConnection=100
initialConnection=10
poolGrowth=5
connectionTimeout=1800000
cleanPoolInterval=600000
maxSize=50
purgeCacheSize=5

#-------------------------------------------------------------------------
# JDBj Runtime properties
#
# These are all optional.  They correspond to system runtime properties
# applicable to JDBj logical connections.
#
# NOTES: * In each case, the values listed are the defaults. The
#          template default entries are commented out.  Make sure
#          and remove the pound sign (#) when changing these values.
#-------------------------------------------------------------------------

[JDBj-RUNTIME PROPERTIES]
as400PackageLibrary=QRECOVERY
 ; Above line added for V3T 3/3/04, not needed for AIO-native driver
 ; Data cache entry size threshold property. Sets the maximum
 ; size (in rows) for a single entry in the data cache. If a
 ; potential entry exceeds this threshold, then it will not be
 ; cached. -1 indicates that all eligible entries should be
 ; cached.
 ;
 ; Valid values: -1 or greater

# dataCacheEntrySizeThreshold=500

 ; Data cache entry threshold property. Sets the maximum size (in
 ; entries) for the data cache. If the data cache exceeds this
 ; size, then the least recently used entries are removed to
```

```
 ; make room for the new entry. -1 indicates that there is no
 ; maximum.
 ;
 ; Valid values: -1 or greater

 ; dataCacheEntryThreshold=100

 ; Force single data source property. Indicates whether JDBj
 ; should force operations involving multiple database objects to
 ; run against a single data source. If this is true, JDBj will
 ; only resolve the physical data source for the first database
 ; object involved in a given operation. Otherwise, JDBj will
 ; resolve physical data sources for all database objects and
 ; throw an exception if a single operation spans multiple data
 ; sources.
 ;
 ; Valid values: true, false

# forceSingleDataSource=false

   ; No where paren property.  Certain TableIO system function
   ; queries require a long list of selections in the where clause
   ; without any intermediate parentheses. Specifying a composite
   ; selections alone for this case creates intermediate
   ; parentheses. If this property is set to a true then the JDBj
   ; will create SQL without any intermediate parenetheses.
   ;
   ; Valid values: true, false

# noWhereParen=false

 ; Result set timeout property. Sets the the number of
 ; milliseconds for the result set timeout. A result set will
 ; timeout if it has not had any operations for a period of time.
 ;
 ; Valid values: -1 or greater

# resultSetTimeout=60000

 ; Retry interval property. Sets the interval (in milliseconds)
 ; to wait before retrying idempotent database operations. -1
 ; means not to retry, 0 means retry immediately.
 ;
 ; Valid values: -1 or greater

# retryInterval=0

 ; Retry maximum property. Sets the number of times to retry
 ; idempotent database operations. -1 means to retry
 ; indefinitely.
 ;
 ; Valid values: -1 or greater

# retryMaximum=5

 ; Spec consistency check property. Sets the level of spec
 ; consistency checking. Set this property to a higher level to
 ; improve exception messages relating to spec inconsistency
 ; problems. Set this to a lower level when it is not needed
 ; since it may degrade performance.
```

```
 ;
 ; Valid values: full, minimal, none

# specConsistencyCheck=minimal

 ; Transaction isolation property. Sets the isolation level to be
 ; used for transactions.
 ;
 ; Valid values: default, none, read uncommitted, read committed,
 ; repeatable read, serializable

# transactionIsolation=default

 ; Transaction timeout property. Sets the number of milliseconds
 ; for the transaction timeout. A transaction will timeout if it
 ; has not had any operations for a period of time. If this
 ; timeout is set to a -1 then the transaction timeout feature
 ; will be disabled.  (This currently only happens when JDBj is
 ; run as part of a JAS transaction.)
 ;
 ; Valid values: -1 or greater

# transactionTimeout=120000

 ; Trigger auto fetch property. Indicates whether JDBj should
 ; automatically fetch rows to be changed in order to pass them
 ; to OneWorld triggers. This is needed for certain triggers to
 ; work properly.
 ;
 ; Valid values: true, false

# triggerAutoFetch=false

 ; Updatable result set timeout property. Sets the the number of
 ; milliseconds for the updatable result set timeout. An
 ; updatable result set will timeout if it has not had any
 ; operations for a period of time. -1 indicates that updatable
 ; result sets should have the same value as RESULT_SET_TIMEOUT.
 ;
 ; Valid values: -1 or greater

# updatableResultSetTimeout=-1

 ; Usage execution threshold property. Sets the expected maximum
 ; time (in milliseconds) for a single database operation to
 ; execute. If a single database operation takes longer than this
 ; threshold, then the statement and time will be logged as part
 ; of usage tracking.
 ;
 ; Valid values: 0 or greater

# usageExecutionThreshold=20000

 ; Usage fetch size threshold property. Sets the expected maximum
 ; fetch size (in rows) for a single fetch to return. If a single
 ; fetch returns more than than this threshold, then the actual
 ; fetch size will be logged as part of usage tracking. -1
 ; indicates that any fetch size is valid.
 ;
 ; Valid values: -1 or greater
```

```
# usageFetchSizeThreshold=500

 ; Usage result set open threshold property. Sets the expected
 ; maximum time (in milliseconds) for a result set to remain
 ; open. If a result set is left open longer than this threshold,
 ; then the result set and time will be logged as part of usage
 ; tracking. This does not automatically close the result set. A
 ; usage result set open threshold value of -1 indicates that no
 ; such usage tracking will occur.
 ;
 ; Valid values: -1 or greater

# usageResultSetOpenThreshold=60000

 ; Usage tracking property. Enables or disables usage tracking.
 ; This provides additional tracking and logging to use during
 ; testing. Usage tracking does incur a performance penalty so it
 ; should be disabled for production code.
 ;
 ; Valid values: true, false

# usageTracking=false

 ; Usage tracking output property. Specifies the destination of
 ; usage tracking output.
 ;
 ; Valid values: log, exception

# usageTrackingOutput=log

 ; Usage transaction active threshold property. Sets the expected
 ; maximum time (in milliseconds) for a transaction to remain
 ; active. If a transaction is left active longer than this
 ; threshold, then the transaction and time will be logged as
 ; part of usage tracking. This does not automatically rollback
 ; the transaction. A usage transaction active threshold value of
 ; -1 indicates that no such usage tracking will occur.
 ;
 ; Valid values: -1 or greater

# usageTransactionActiveThreshold=120000

#============================================================================
#                              END OF FILE
#============================================================================
```

The jde.ini file

Example B-4 lists the contents of the jde.ini file.

Example: B-4 The jde.ini file

```
#==========================================================================
#                    jde.ini file
#
; f:\B9SYS\INI\jde.ini
; Enterprise jde.ini on denis1a
; OneWorld initialization file INI(JDE)
; AS/400 specific version - B9

[JDE_CG]
CLASSPATH=/B9SYS/classes/xalan.jar:/B9SYS/classes/xerces.jar:/B9SYS/classes/kernel.jar:/B9S
YS/classes/XTS.jar:/B9SYS/classes/log4j.jar:/B9SYS/classes

[DEBUG]
Output=NONE
DebugFile=/JDEB9/JDEDEBUG
;DebugFile=/JDEB9/JDE.LOG
JobFile=/JDEB9/JDE.LOG
JDETSFile=/JDEB9/JDETS.LOG
LogErrors=1
ClientLog=0
KeepLogs=1
RunBatchDelay=0
TAMTraceLevel=0

[JDEMAIL]
Rule1=90|OPT|MAILSERVER=mail.jdedwards.com
Rule2=100|DEFAULT|OWMON=OWMON@jdedwards.com
Rule3=110|DEFAULT|JDE_SYSTEM=JDE_System@jdedwards.com
Rule4=120|DEFAULT|WORKFLOW_SYSTEM=Workflow@jdedwards.com
Rule5=130|OPT|MERGELOCAL=1
Rule6=140|OPT|UPDATELOCAL=0
ClientType=Windows

[SVR]
EnvironmentName=PD9
LibraryListName=PD9
SpecPath=specfile
ApplicationPathAddendum=

[INSTALL]
;POSTSCRIPT_ONLY=1
DefaultSystem=B9SYS
B9=
Double_Byte=0
LocalCodeSet=US_EBCDIC

[JDEIPC]
maxNumberOfResources=1000
startIPCKeyValue=2101
avgResourceNameLength=15
maxMsgqEntries=1024
maxMsgqBytes=65536
ipcTrace=0
```

```
[JDENET]
serviceNameListen=6011
serviceNameConnect=6011
maxNetProcesses=2
maxNetConnections=800
netShutdownInterval=15
maxKernelProcesses=100
maxKernelRanges=24
netTrace=0
enablePredefinedPorts=0
checkKrnlHealth=150000

[JDENET_KERNEL_DEF1]
krnlName=JDENET RESERVED KERNEL
dispatchDLLName=JDENET
dispatchDLLFunction=JDENET_DispatchMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF2]
krnlName=UBE KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchUBEMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=1

[JDENET_KERNEL_DEF3]
krnlName=REPLICATION KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=DispatchRepMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF4]
krnlName=SECURITY KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchSecurity
maxNumberOfProcesses=2
numberOfAutoStartProcesses=2

[JDENET_KERNEL_DEF5]
krnlName=LOCK MANAGER KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=TM_DispatchTransactionManager
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF6]
krnlName=CALL OBJECT KERNEL
dispatchDLLName=XMLCALLOBJ
dispatchDLLFunction=XMLCallObjectDispatch
maxNumberOfProcesses=14
numberOfAutoStartProcesses=14


[JDENET_KERNEL_DEF7]
krnlName=JDBNET KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchJDBNETMessage
maxNumberOfProcesses=1
```

```
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF9]
krnlName=SAW KERNEL
dispatchDLLName=JDESAW
dispatchDLLFunction=JDEK_DispatchSAWMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF10]
krnlName=SCHEDULER KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchScheduler
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF11]
krnlName=PACKAGE BUILD KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchPkgBuildMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF12]
krnlName=UBE SUBSYSTEM KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=JDEK_DispatchUBESBSMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF13]
krnlName=WORK FLOW KERNEL
dispatchDLLName=WORKFLOW
dispatchDLLFunction=JDEK_DispatchWFServerProcess
maxNumberOfProcesses=5
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF14]
krnlName=QUEUE KERNEL
dispatchDLLName=JDEKRNL
dispatchDLLFunction=DispatchQueueMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF15]
krnlName=XML TRANS KERNEL
dispatchDLLName=XMLTRANS
dispatchDLLFunction=XMLTransactionDispatch
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF16]
krnlName=XML LIST KERNEL
dispatchDLLName=XMLLIST
dispatchDLLFunction=XMLListDispatch
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF19]
krnlName=EVN KERNEL
```

```
dispatchDLLName=JDEIE
dispatchDLLFunction=JDEK_DispatchITMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF20]
krnlName=IEO KERNEL
dispatchDLLName=JDEIEO
dispatchDLLFunction=JDEK_DispatchIEOMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF22]
krnlName=XML DISPATCH KERNEL
dispatchDLLName=XMLDSPATCH
dispatchDLLFunction=XMLDispatch
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF23]
krnlName=XTS KERNEL
dispatchDLLName=XTSKRNL
dispatchDLLFunction=JDEK_DispatchXTSMessage
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[JDENET_KERNEL_DEF24]
krnlName=XML SERVICE KERNEL
dispatchDLLName=XMLSERVICE
dispatchDLLFunction=XMLServiceDispatch
maxNumberOfProcesses=1
numberOfAutoStartProcesses=0

[DB SYSTEM SETTINGS]
Version=43
Default User=JDE
Default Pwd=JDE
Default Role=*ALL
Default Env=PD9
Default PathCode=PD9
Base Datasource=DENIS1A - B9 Server Map
Object Owner=
Server=DENIS1A
Database=DENIS1A - B9 Server Map
Load Library=
Decimal Shift=Y
Julian Dates=Y
Use Owner=N
Secured=Y
Type=I
Library=
Library List=
DatabaseName2=SVM9
DatabaseInstance=
ServerPort=
JDBNETUse=N
UnicodeFlag=Y
LOBFlag=Y
DatabaseProgramMax=-1
DatabaseProgramInitial=10
```

```
DatabaseProgramThreshold=3
DatabaseProgramAdditional=10
DatabaseProgramCheckIntervalSeconds=10
Default Journal=OW_JRNL
Default Journal LIBRARY=OWJRNL
Default Journal Receiver=OW_JRNL000
Default Journal Receiver LIBRARY=OWJRNL

[NETWORK QUEUE SETTINGS]
DefaultPrinterOUTQ=QGPL/ONEWORLD_A
QKActive=0
QKOnIdle=300
JDENETTimeout=300
#Added 6/25/04.  The default is 30 sec. which is too low for high transaction systems.

[BSFN BUILD]
BuildArea=/OneWorld/Packages
OptimizationFlags=(40)
DebugFlags=*ALL
InliningFlags=Y
DefineFlags=JDENV_AS400MUTEX PRODUCTION_VERSION JDBDB2400 AS400V3R6
CompilerFlags=*EXPMAC *NOSHOWINC
CompileOutput=*PRINT
OSReleaseLevel=*CURRENT
LinkFlags=*DUPPROC *DUPVAR *UNRSLVREF
LinkLibraries=JDEUNICODE JDELIB JDEKRNL OWVER V_VERIFY XERCESWRAP
#added xerceswrap for sp2 12/17/03
#added below for server package build 8 threads
QNAME=QBATCH8
SimultaneousBuilds=5

[AS400]
CRTCMOD=CRTCMOD MODULE(%ls/%ls) SRCFILE(%ls/%ls) SRCMBR(%ls) OUTPUT(*PRINT) DBGVIEW(*NONE)
OPTIMIZE(40)
CRTCMOD2=DEFINE(JDENV_AS400MUTEX PRODUCTION_VERSION NO_SIGNALS) TGTRLS(*CURRENT)
CRTSRVPGM=CRTSRVPGM SRVPGM(%ls/%ls) MODULE(%ls/*ALL) BNDSRVPGM(JDEUNICODE JDELIB JDEKRNL
OWVER) EXPORT(*ALL) OPTION(*DUPPROC *DUPVAR *UNRSLVREF) ALWLIBUPD(*YES) TGTRLS(*CURRENT)
CRTDBPGM1=CRTPGM PGM(%ls/%ls) MODULE(DBDRVAG DBDRV DBDRV_AC DBDRV_CC DBDRV_CN
CRTDBPGM2=DBDRV_CH DBDRV_CP DBDRV_DT DBDRV_RQ DBDRVSQL DBMONCTL DBDRVCLI
CRTDBPGM3=DBSQL DBSQL_A DBSQL_D DBSQL_I DBSQL_M DBSQL_S DBSQL_U DBSQL_X)
CRTDBPGM4=BNDSRVPGM(JDEUNICODE JDEKRNL JDELIB JDEIPC QBFCPRCED) ACTGRP(%ls) OPTION(*DUPPROC
CRTDBPGM5=*DUPVAR) ALWLIBUPD(*YES) AUT(*ALL) TGTRLS(*CURRENT)
PrintUBEJoblog=FALSE
PrintUBEJoblogOnError=FALSE

[SERVER ENVIRONMENT MAP]
JDEPLAN=JD9

[UBE]
UBEDebugLevel=0

[SECURITY]
DataSource=System - B9
User=JDE
Password=JDE
Default Role=*ALL
DefaultEnvironment=PD9
SecurityServer=DENIS1A
ServerPswdFile=TRUE
History=0
```

```
[LOCK MANAGER]
Server=
AvailableService=NONE
RequestedService=None

[TCENGINE]
TraceLevel=1
StopAfterRow=0
ForceRowByRow=0

[INTEROPERABILITY]
FilteredEvents=*NONE
RegisteredEvents=

[SAMPLE_EVENT]
DS1=DXXXXXXXX
DS2=DYYYYYYYY
DS3=DZZZZZZZZ

[JDEITDRV]
DrvCount=4
Drv1=RT:RTDRV
Drv2=WF:WFDRV
Drv3=Z:ZDRV
Drv4=JDENET:JDETRDRV

[LRENGINE]
System=/B9SYS/output
Repository_Size=20
Disk_Monitor=YES

[WORLD ENVIRONMENT MAP]

[WORKFLOW]
; Workflow Runtime Interop, inbound (server only)
WRIUser=xxx
WRIPassword=xxx
WRIEnvironment=xxx
WRIRole=*ALL
; Workflow Runtime Interop, outbound
PrimaryEvnServer=xxx
SecondaryEvnServer=xxx

[JAS PREFERENCE]
JasServer=
Port=80
Servlet=/jde/servlet/html.login

;[TRANSPORTS]
;TransportName1=xxxTransportExeNamexxx
;TransportParams1=xxxTransportParametersxxx

[XAPI]
XMLDirectory=JDEB9

[XMLLookupInfo]
XMLRequestType1=list
XMLKernelMessageRange1=5257
XMLKernelHostName1=local
```

```
XMLKernelPort1=0

XMLRequestType2=callmethod
XMLKernelMessageRange2=920
XMLKernelHostName2=local
XMLKernelPort2=0

XMLRequestType3=trans
XMLKernelMessageRange3=5001
XMLKernelHostName3=local
XMLKernelPort3=0

XMLRequestType4=JDEMSGWFINTEROP
XMLKernelMessageRange4=4003
XMLKernelHostName4=local
XMLKernelPort4=0
XMLKernelReply4=0

XMLRequestType5=xapicallmethod
XMLKernelMessageRange5=14251
XMLKernelHostName5=local
XMLKernelPort5=0

XMLRequestType6=realTimeEvent
XMLKernelMessageRange6=14251
XMLKernelHostName6=local
XMLKernelPort6=0
XMLKernelReply6=0

[XTSRepository]
XTSRepositoryPath=/B9SYS/XtsRepository
XSL-JDE-BIRDS-REQUEST=birds.xsl
XSL-JDE-V2_2_NATIVE-REQUEST=v2toNative.xsl
XSL-JDE-V1_NATIVE-REQUEST=ml.xsl
XSL-JDE-V1_NATIVE-RESPONSE=lm.xsl

[XTS]
XTSTemplateSelector1=com.jdedwards.xts.xtsm.XTSMJDETemplateSelector
XTSTraceLevel=2
XTSDelay=0

[MTR VALIDATION]
EnterpriseDiskInMB=5000

#=========================================================================
#                              END OF FILE
#=========================================================================
```

C

Additional information for switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment

This appendix provides additional information for Chapter 7, “Switched disk cluster topology for a JD Edwards EnterpriseOne 8.0 environment” on page 85.

© Copyright IBM Corp. 2005. All rights reserved.

Relational database directory entry

The relational database directory (RDB) entry might be a new concept for many iSeries users. On an iSeries system without an IASP there is only one database and therefore only one RDB entry. When an IASP is created, the system adds an additional RDB entry with the same name as the IASP automatically.

Use the Work With Relational Database Directory Entries command to look at the RDB entries on a system:

```
WRKRDBDIRE *all
```

The Work With Relational Database Directory Entries screen appears as shown in Figure C-1.

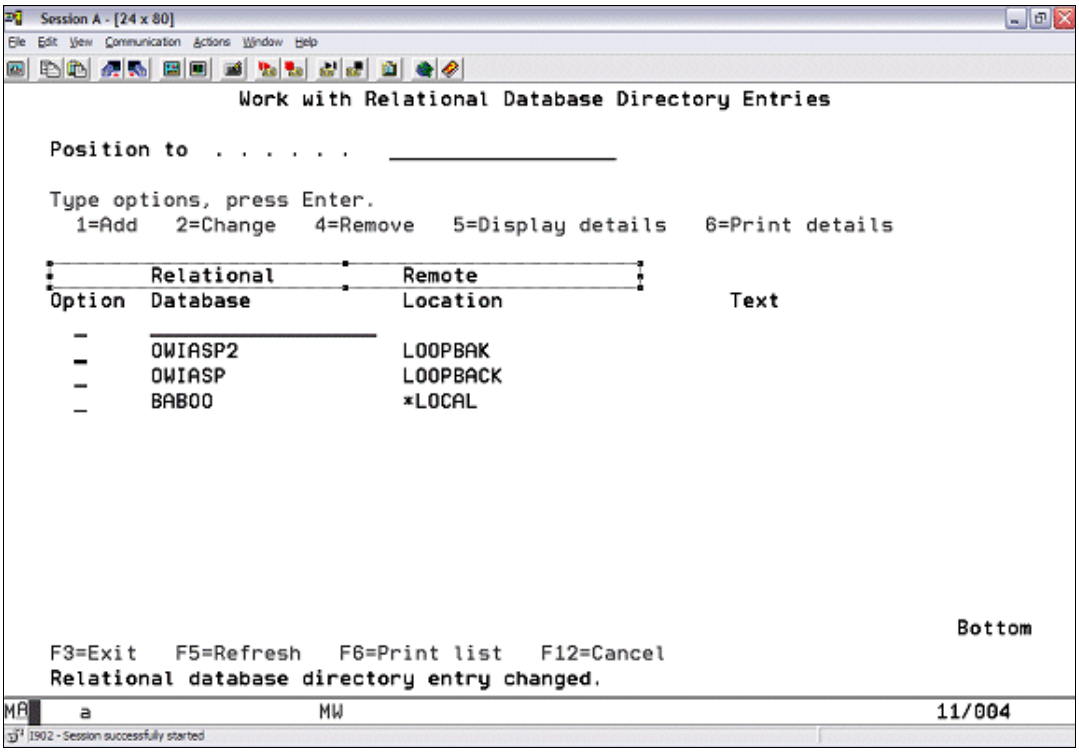

Figure C-1 Work with Relational Database Directory Entries

As illustrated in Figure C-1, there are two different kinds of RDB entries:

- One entry with the remote location as `*LOCAL`
- Multiple entries with the remote location as `LOOPBACK`.

The RDB entry with the remote location as *LOCAL is associated with system space (system ASP). The system only allows one *LOCAL RDB entry. However it is possible to have several LOOPBACK RDB entries. This is because it is possible to have more than one IASP on a given system.

Use the Display Device Description command to see which LOOPBACK RDB entry is associated with an IASP:

```
dspdevd name of IASP
```

The Display Device Description screen appears as shown in Figure C-2.

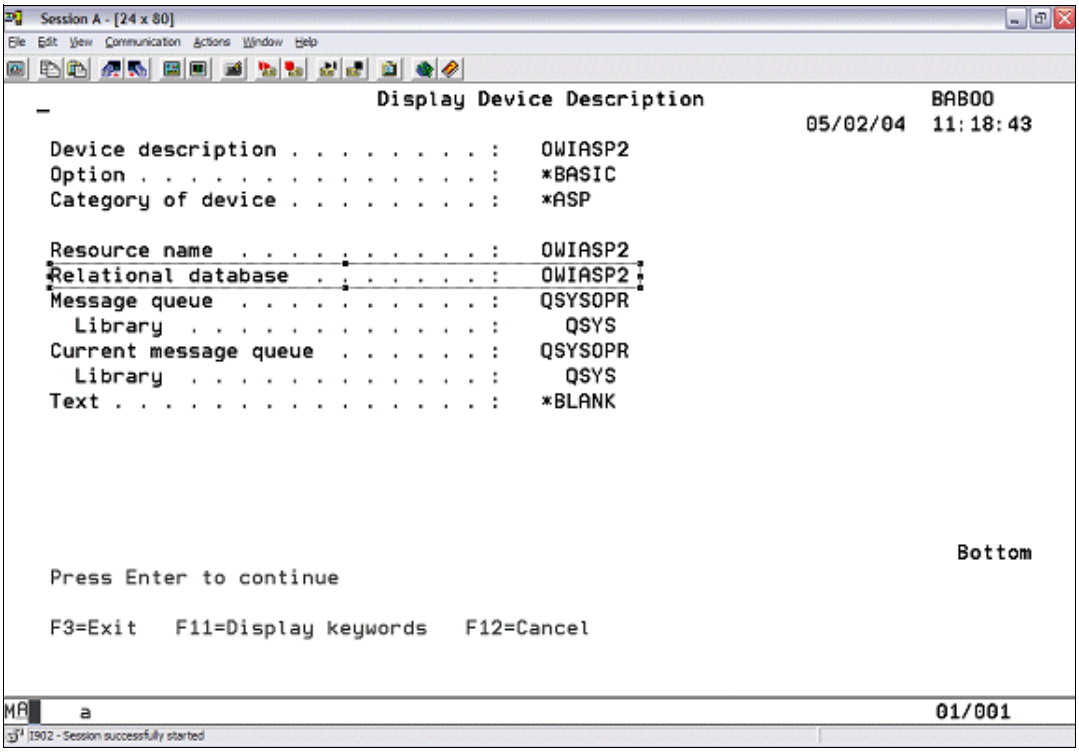

Figure C-2 Display Device Description

The IASP in Figure C-2 is named OWIASP2. It is associated with RDB entry OWIASP2.

Synchronizing user profiles across nodes

This section provides step-by-step instructions on how to synchronize user profiles across nodes using iSeries Navigator.

Begin by collecting an inventory of groups and users in the cluster as shown in Figure C-3.

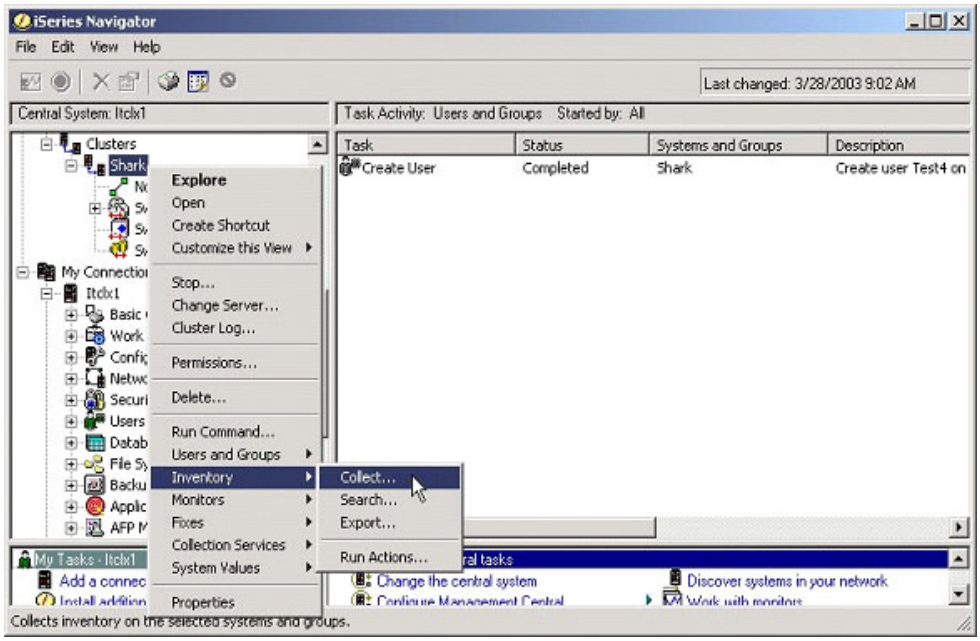

Figure C-3 Collecting inventory

Select **Users and Groups** for the type of inventory to collect, then click **OK**. Verify the task completed successfully by selecting **Inventory** under **Task Activity**.

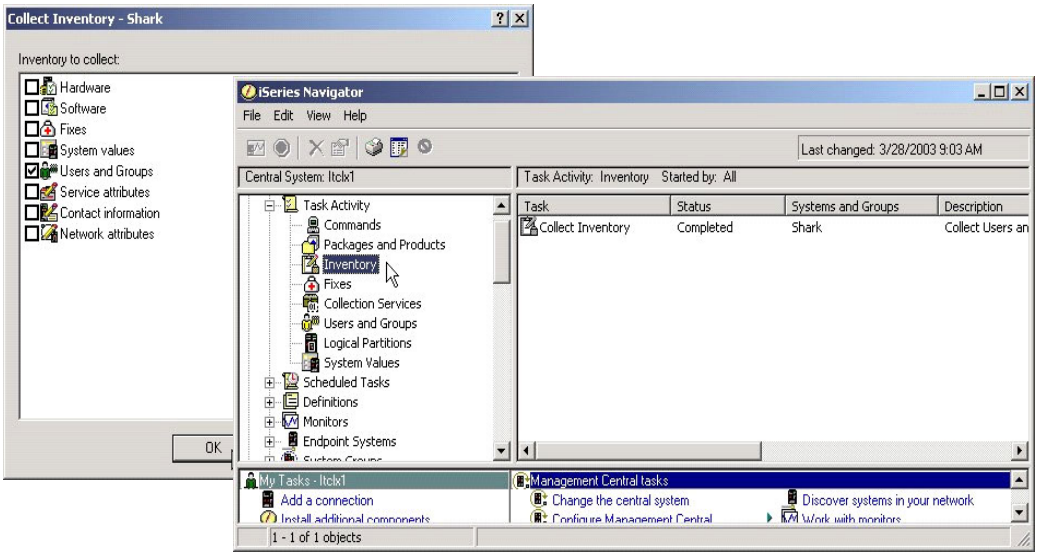

Figure C-4 Inventory

Next, select **Edit Users** under **Users and Groups** for the cluster.

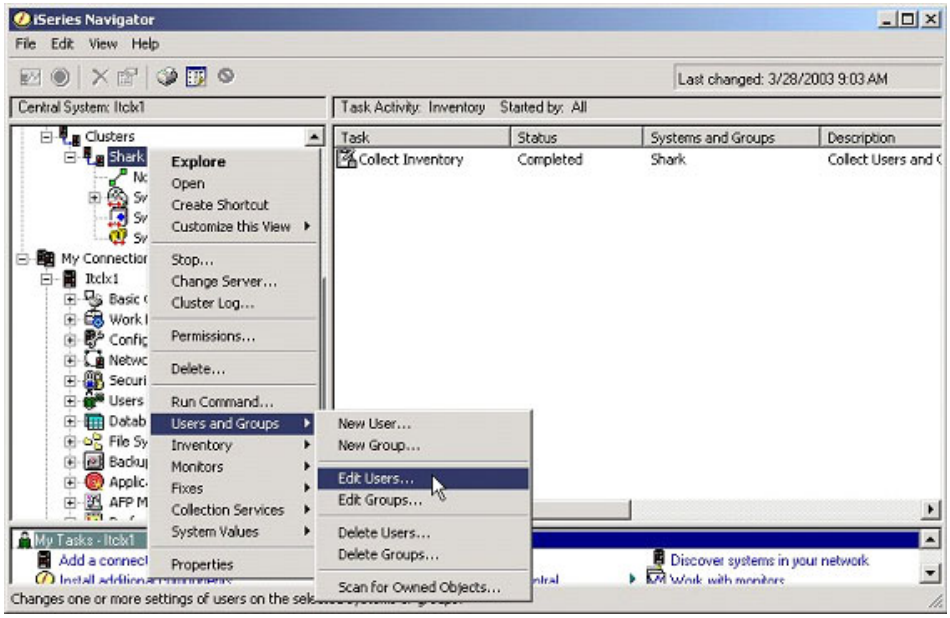

Figure C-5 Edit Users

Click **Browse**. Select the user profiles to synchronize, and then click **OK**.

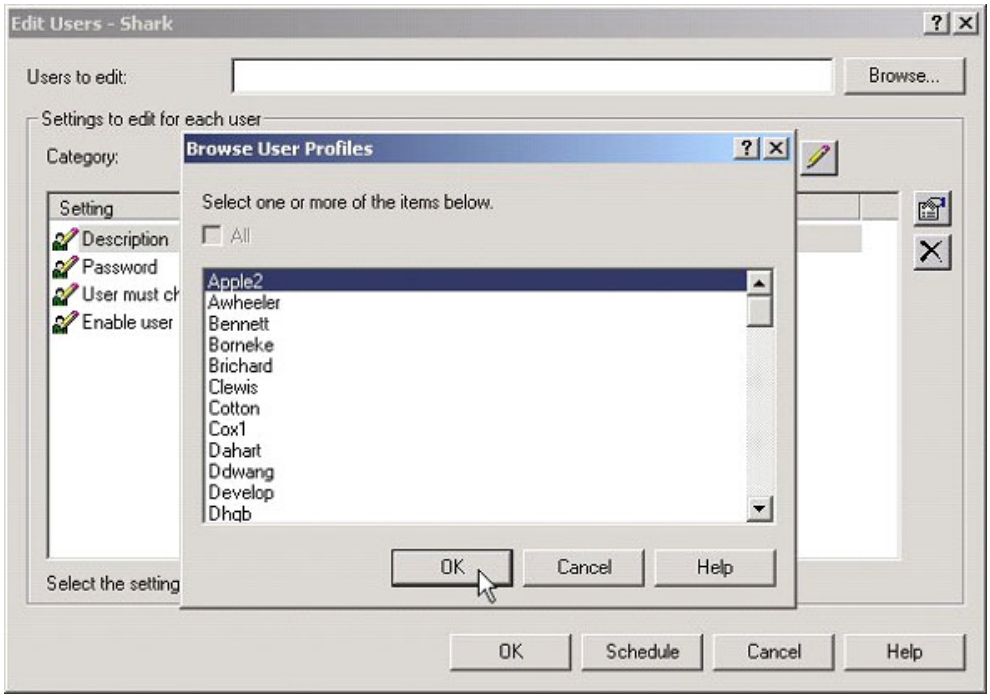

Figure C-6 Select user profiles to synchronize

Select the category to edit (Unique Identifier), and then click **Properties**.

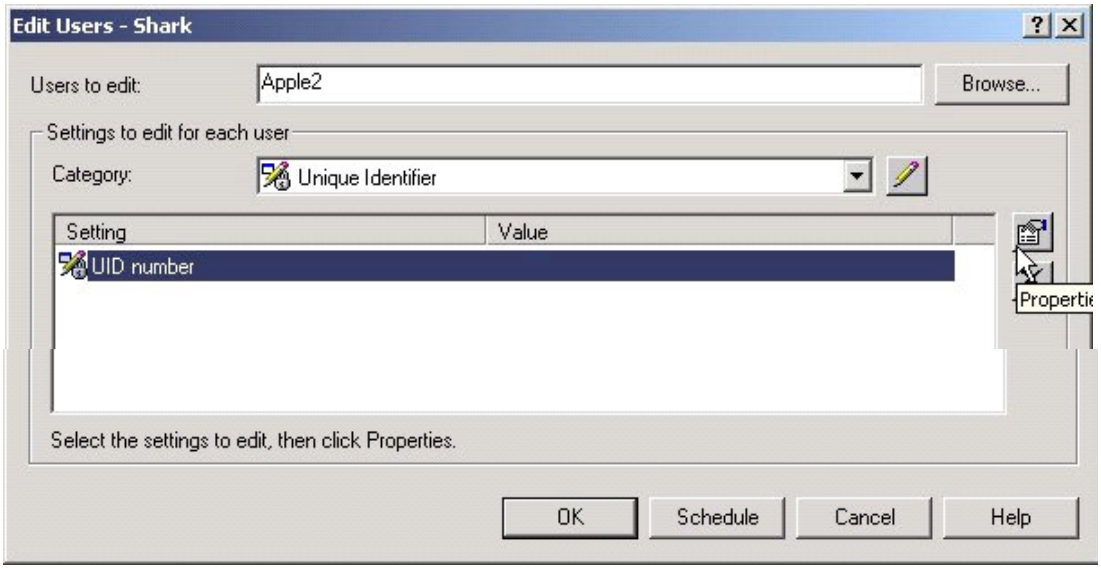

Figure C-7 Select category to edit

Select **Find a unique identifier across all selected systems**, then click **OK**.

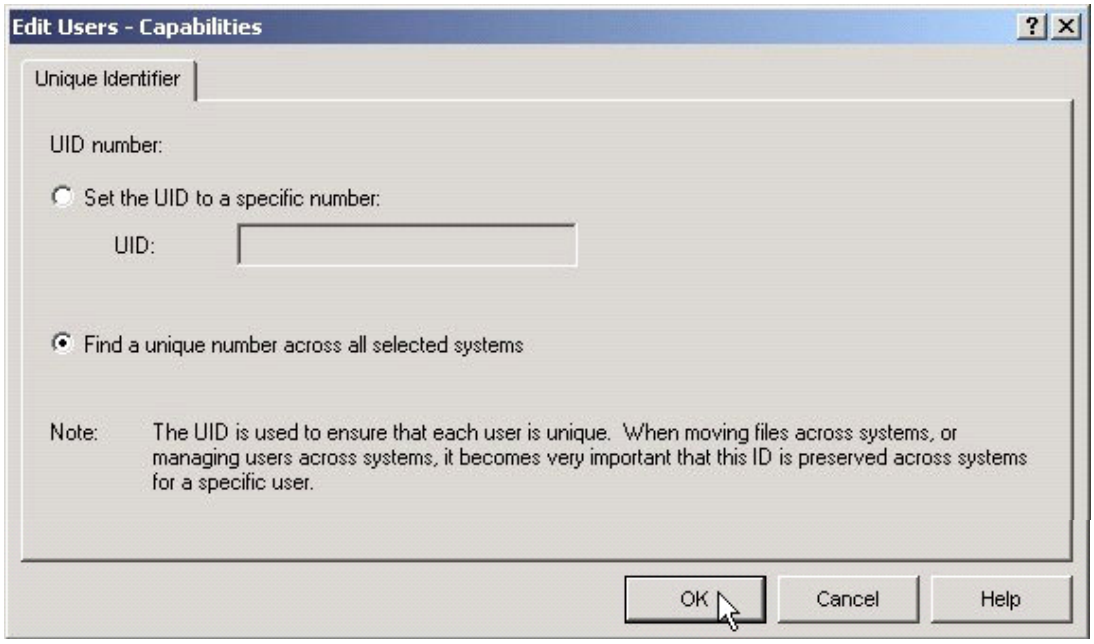

Figure C-8 Specify unique ID

Click **OK** to synchronize the user profile on both nodes.

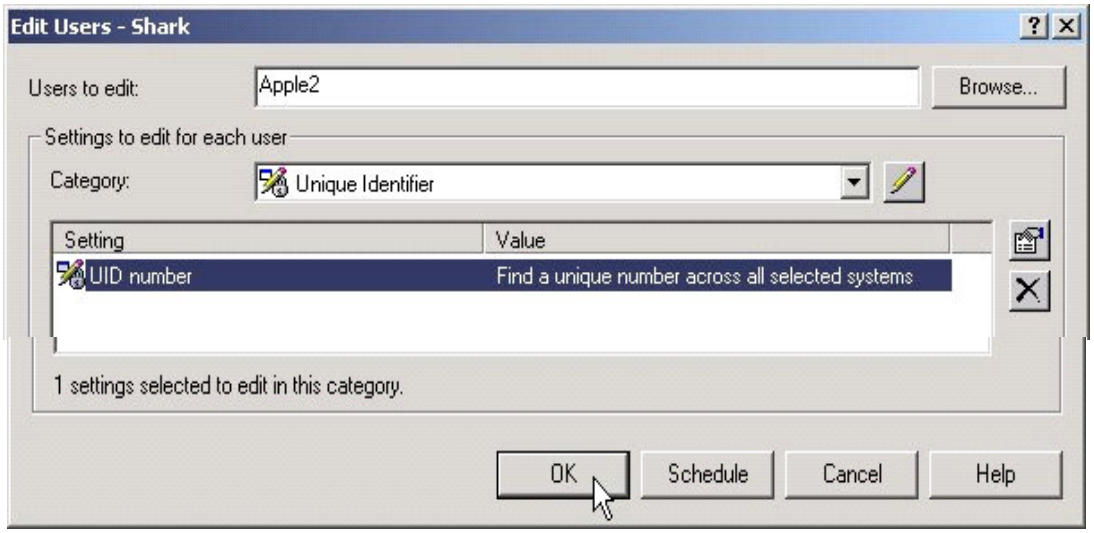

Figure C-9 Synchronize user profile

Verify that the task completed successfully. If it did not, double-click it to investigate.

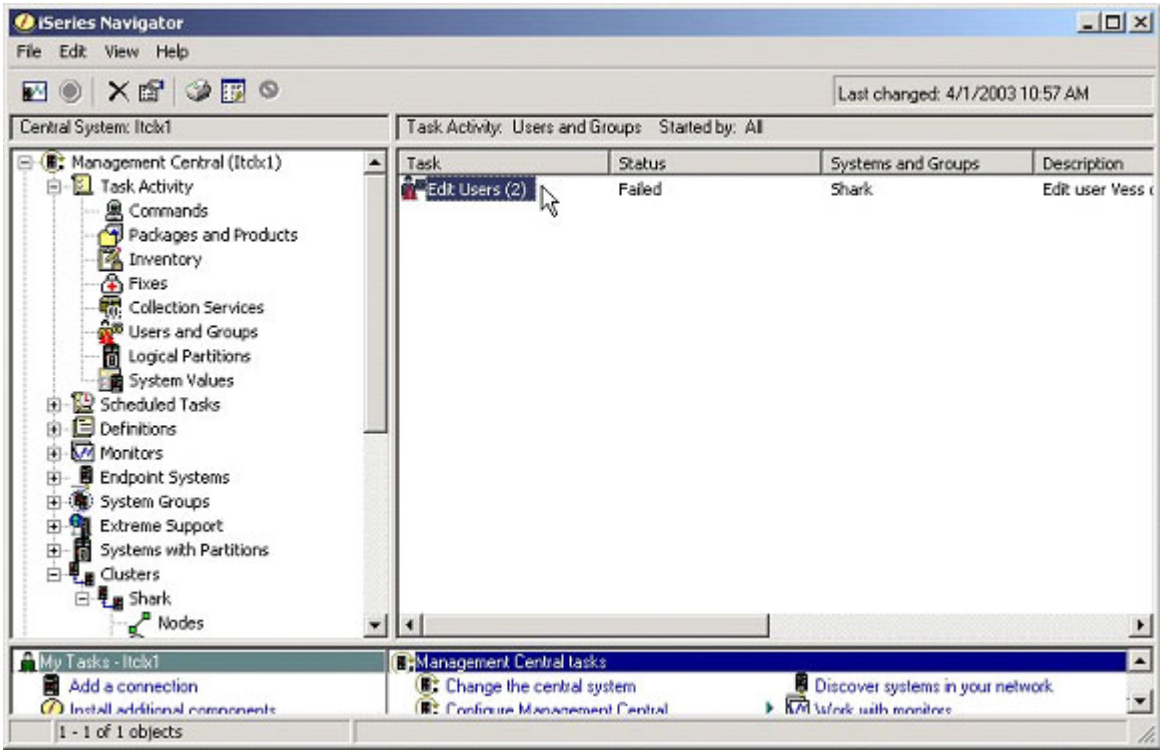

Figure C-10 Verify completion of task

Often, the cause of the error is apparent on the first status screen (in this case, the user profile is in use when synchronization was attempted). If the cause of the error is not clear, right-click each node to examine the job log on that system.

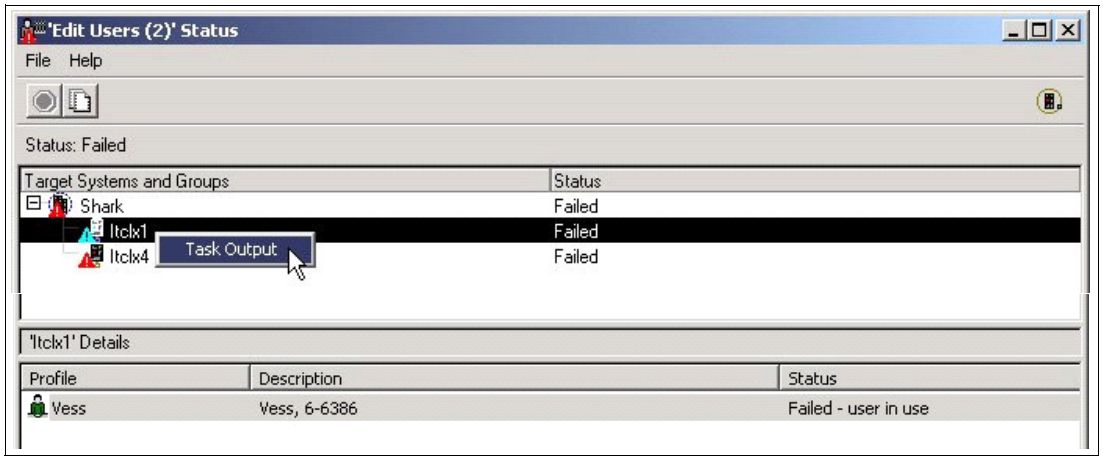

Figure C-11 Status

Related publications

This section lists the publications that are considered particularly suitable for a more detailed discussion of the topics that are covered in this redbook.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 155. Note that some of the documents referenced here might be available in softcopy only.

- *Using AS/400 Database Monitor and Visual Explain to Identify and Tune SQL Queries*, REDP-0502
- J.D. Edwards OneWorld Implementation for AS/400, SG24-5195
- *Introduction to Storage Area Network (SAN)*, SG24-5470
- *AS/400 Disk Storage Topic and Tools*, SG24-5693
- *Designing an IBM Storage Area Network*, SG24-5758
- *Storage Consolidation in SAN Environments*, SG24-5987
- *Consolidating Windows 2000 Servers in iSeries: An Implementation Guide for the IBM Integrated xSeries Server for iSeries*, SG24-6056
- *Implementing an Open IBM SAN*, SG24-6116
- *The IBM SAN Survival Guide*, SG24-6143
- *IBM @server iSeries in Storage Area Networks: A Guide to Implementing FC Disk and Tape with iSeries*, SG24-6220
- *Direct Attach xSeries for the IBM iSeries Server: A Guide to Implementing xSeries Servers in iSeries*, SG24-6222
- *Introduction to SAN Distance Solutions*, SG24-6408
- *J.D. Edwards OneWorld Xe Implementation on IBM @server iSeries Servers*, SG24-6529
- *Microsoft Windows Server 2003 Integration with iSeries*, SG24-6959
- *IBM @server iSeries Independent ASPs: A Guide to Moving Applications to IASPs*, SG24-6802
- *Logical Partitions on IBM PowerPC: A Guide to Working with LPAR on POWER5 for IBM @server i5 Servers*, SG24-8000

Other publications

These publications are also relevant as further information sources:

- *IBM TotalStorage Enterprise Storage Server User's Guide*, SC26-7295
- *IBM TotalStorage Enterprise Storage Server Host Systems Attachment Guide*, SC26-7296
- *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide*, SC25-7346
- *IBM TotalStorage Enterprise Storage Server Quick Configuration Guide*, SC26-7354

© Copyright IBM Corp. 2005. All rights reserved.

- *JD Edwards EnterpriseOne Release 8.9 Installation PeopleBook for iSeries-Based System*
- *JD Edwards Learning Management 8.81 Self-service using Oracle9i in a mixed Hewlett-Packard PA-RISC/Itanium UNIX Environment*, 2004, at:

  http://www.peoplesoft.com
- *Linux Scalability Center. Samba Fileserving on IBM @server zSeries servers with z/VM*, June 2003, at:

  http://www.ibm.com/servers/eserver/zseries/library/whitepapers/gm130250.html
- *WebSphere 5.0 Performance Sample Application: Trade3*, available with Trade3 download at:

  http://www-306.ibm.com/software/webservers/appserv/benchmark3.html
- *ESS FICON Channel Attachment Version 5.01* white paper at

  http://www.storage.ibm.com/hardsoft/products/ess/support/essfcwp.pdf
- Hata, Dan. *Enterprise Learning Management Benchmark Using Oracle9i on HP Hardware*, January 2004, at:

  http://www.peoplesoft.com
- T*he IBM @server i5 Three-in-One Benchmark*, May 2004, at:

  http://www.ibm.com/servers/eserver/iseries/hardware/threeinone

Online resources

These Web sites and URLs are also relevant as further information sources:

- The System Administration Workbench

  http://<sys>:<port>/jde/saw/sawJas.html
- Oracle

  http://www.peoplesoft.com/corp/en/public_index.jsp
- Recommended PTF's and Informational APAR for JD Edwards

  http://www-1.ibm.com/servers/enable/site/events/jde_links.html
- Minimum Technical Requirements (MTR) document

  http://www.peoplesoft.com/corp/en/public_index.jsp
- iSeries Technology Center Web site with educational opportunities and implementation assistance

  http://www.ibm.com/servers/eserver/iseries/service/itc/index.html
- High Availability solutions for the iSeries server

  http://www.ibm.com/servers/eserver/iseries/ha/
- iSeries System Performance Team. *The IBM @server i5 Three-in-One Benchmark*. May 2004.

  http://www.ibm.com/servers/eserver/iseries/hardware/threeinone
- OS/400 and hardware specifications regarding Windows Integration

  http://www-1.ibm.com/servers/eserver/iseries/windowsintegration/
- Complete and current information about servers, operating systems, host adapters, and connectivity products supported by the ESS

  http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm

- Introduction to terminology and benefits of SAN storage, and a description of the SAN initiative from IBM:

  http://www.storage.ibm.com/ibmsan/basics.htm

- ESS Interoperability Matrix and PPRC requirements

  http://www.storage.ibm.com/disk/ess/pdf/interop.pdf

- *ESS FICON Channel Attachment Version 5.01* white paper

  http://www.storage.ibm.com/hardsoft/products/ess/support/essfcwp.pdf

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications, and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Symbols
*SAVF 70

Numerics
5722-DG1 26
5722-PT1 26
5722-WS5 35
5722-WSV 67
5733-WS5 35

A
access log 26
  disabling 30
access logging 26, 28
activity level 39
advanced parameter 56
AIO 62
AIX 5L 71
all-in-one environment (AIO) 120
Apache HTTP Server instance 26
application
  P986130 56
  Trade3 77
application software
  JD Edwards Enterprise One 8.9 74
application tuning 41
application workload
  Enterprise Learning Management 8.81 74
  Trade3 74
asynchronous cascading (PPRC) 110

B
backup window 117
benchmark 72, 75
  virtualization 119, 121
  Web Server 119
benchmark results 81
benefits of external storage 116
binary large object (BLOB) 56
browser interface 26
byte counts 27

C
cache 114
cache miss ratio 37
calculating size of deployment server 64
call object kernel jobs 42–43
CCSID 60
CD
  central object 14
  first 9
  fourth 9
  host code 14
  install 9
  setup 9
  third 9
CDROM
  lock 69
central objects 9
  CDs 14
  loading 15
change shared pool 114
check kernel health 43
checklist
  pre-installation 66
CHGSHRPOOL 114
cluster resource group 86
code page 55
communication workload 27
completing the install 23
compression 32
configuration
  dedicated SAN 114
  IXS 66
  network 77
  WebSphere 125
configuration file 26
connectivity 115
create a job description 95
create Network Server Storage space 69
create save file 70
creating DB indexes 47
CRG 86

D
data compression 32
data file 9
data source table 93
database connection 49
database connection type 45
database index
  creating 47
database tuning 44
DB2 Universal Database 44
dedicated SAN configuration 114
deployment server 8
  calculating size 64
  vary on 69
Director Multiplatform Agent 78
directory
  IASP 90
  root 91
directory directives 34
disabling the access log 30
disaster recovery 116
disk drive module (DDM) 110

© Copyright IBM Corp. 2005. All rights reserved.

display system value 4
distributed I/O processor 113
DNS 86
Domain Name Server 86
DSPSYSVAL 4
DVD
  lock 69

E

edit job description 96
edit user profiles 96
EDRS 45
EDRSQL 2nd connection 8
Electronic Software Update (ESU) 13
Enterprise Learning Management 77
Enterprise Learning Management workload 73
EnterpriseOne Installation Workbench 16
EnterpriseOne printers 22
EnterpriseOne services
  starting 21
environment
  All-In-One (AIO) 120
  all-in-one (AIO) 120
  software 87
  test 120
  Virtual Three Tier (V3T) 120
EO8.9 iSeries Upgrade.pdf 55
ERP 111
error logging 28
ESS Copy Services 107
ESS Model 800 117
expert cache 111
Extended Dynamic Remote SQL (EDRSQL) server 8
Extended Dynamic Remote Support 45
external DASD 116
external storage
  benefits 116

F

F986101 table 94
F98611 table 94
F98BLOB table 56
fat client machines 98
file
  configuration 26
  data 9
  help 9
  httpd.conf 26, 34, 126
  JAS.INI 101
  jas.ini 42, 127
  JDBBJ.ini 49
  JDBj.ini 51, 130
  jde.ini 42–43, 137
  jdeinst.log 23
  library 18
  Oracle 9
  QAQQINI 4, 57
  save file 70
  SQL server 9
firewall 73
firmware 72
first CD 9
FlashCopy 107
foundation library 18
fourth CD 9
FTP services 8

G

garbage collection 38
global copy 110
GMT 41

H

Hardware Management Console 80
HBA 112
heap size 37–38
help file 9
High Availability solutions 102, 154
HMC 80
Host Bus Adapter 112
host code CDs 14
host table entry 98
HTTP Server Administration interface 26
HTTP Server for iSeries 26
httpd.conf file 26, 34, 126
hub 113
Hypervisor 72

I

IASP 85
IASP directory 90
IBM Director Multiplatform Agent 78
IBM Virtualization Engine Console 72
install
  completing 23
install CDs 9
install iSeries Integration for Windows server 67
installation
  client workstations 22
  completing 23
  custom 16
  customer 16
  Integrated xSeries Adapter for iSeries 63
  Integrated xSeries Server for iSeries 63
  MSDE 10
  scenario
    installation
      typical scenario 16
Installation Workbench 16
instructions for setting up the operating environment 66
INSWNTSVR 67
Integrated xSeries Adapter for iSeries 64
Integrated xSeries Adapter for iSeries installation 63
Integrated xSeries Adapters 71
Integrated xSeries Server 9, 71, 73, 77
Integrated xSeries Server for iSeries 64
Integrated xSeries Server for iSeries installation 63

interactive mode 22
internal storage 116
interoperability, SAN and iSeries 112
IOP 113
iSeries Access for Windows 78
iSeries Performance Monitor 45
iSeries storage 111
iSeries Technology Center web site 102, 154
IXA 63–64, 71
IXS 9, 63–64, 66, 71, 73, 77
  restarting 69
IXS configuration 66

J

JAS 35, 62
JAS network connection 101
jas.ini 101
JAS.INI file 101
jas.ini file 42, 127
Java 111
Java Application Server 62
Java Database Connectivity 45
JavaServer Pages 35
JD Edwards Enterprise Learning Management 8.81 74
JD Edwards Upgrade Guide 55
JDBC 45
JDBC connection 49
JDBj.ini file 49, 51, 130
jde.ini file 42–43
jdeinst.log file 23
jdej.ini file 137
JDENET connections for the Java Server 44
JDEOW 15
job
  call object kernel 42–43
  network kernel 43
  prestart 42, 47
  security kernel 43
  UBE 56
job description 95
  creating 95
  edit 96
JSP 35

K

keep alive 30
kernel settings 42

L

library
  foundation 18
  JDEOW 15
  OWINSTALL 14
  pathcode 18
  QRECOVERY 51
  save file 14
  server 18
  SQL Package 56
library file 18
Link Business Function 19
LINKBSFN 19
Linux 71
Linux 2.6 73
Linux firewall 73
LoadRunner script 73–74
local RDB 147
lock drive 69
log file 23
logging
  access logging 28
  error logging 28
Logical Unit Number (LUN) 110
long distance data copy 110
loopback RDB 147
Lotus Domino 111
LUN 110

M

MAF 50
Management Central 78
maximum users for the Java Server 43
memory pools 38
Mercury QuickTest Professional 73
Metro Mirror 109
metro/global copy 110
Microsoft Database Engine installation 10
Microsoft servers 111
migration 110
Minimum Technical Requirements (MTR) 4
mode
  interactive 22
  push 22
Model 800 117
MSDE installation 10
multi-pathing 115
Multiple Application Framework 50

N

network configuration 77
network kernel job 43
Network Server Storage space 69
network storage space 64

O

Object Configuration Manager 93
OCM 93
OCM table 93
ODBC 45
ODBC settings 99
online transaction processing 111
Open Database Connectivity 45
Oracle file 9
OWINSTALL 14

P

P986130 application 56

paging file size 8
path codes 9
pathcode library 18
peer-to-peer-remote-copy (PPRC) 108
PeopleSoft EnterpriseOne Xe 61
performance 113–114
  application 41
  application server 35
  database 44
  performance management 25
  results 79
  SAN 112
  settings 27
  table conversion 55
  tape device 117
  tests 61
  Unicode conversion 61
  WebSphere Application Server 35
Performance Explorer 45
Performance Monitor 26, 45
Performance Tools for iSeries 26
Planner Validation Workbench 16
planning 66
PORTTEST program 21
POWER Hypervisor 72
PPRC 108
PPRC extended distance 110
PPRC synchronous mode 109
pre-installation checklist 66
prestart job 6, 42, 47
printers 22
processor sharing 72
program
  PORTTEST 21
PTFs 4
push mode 22

Q

QAQQINI file 4, 57
QCCSID system value 59
QRECOVERY library 51
QTP 73
QTP script 74

R

RDB 146
  local 147
  loopback 147
RDBMS 44
real-time statistics 27
reasons to implement SAN technology 112
Redbooks Web site 155
  Contact us xi
references
  IASP 102
Relational Database Directory entry 88, 146
Relational Database Management System 44
release component 9
restarting the IXS card 69
restoring DB libraries to the IASP 90
results 81
  benchmark 81
root directory 91

S

Samba file server 77
Samba file serving 73
Samba workload 76
SAN
  performance 112
  reasons to implement 112
SAN storage cache 114
SAN switch and hub 113
save file 70
save file library 14
save-while-active 107
SAW 35
SAW for enterprise servers 43
script
  /UniDtaConv.xml 60
  LoadRunner 73–74
  QTP 74
  UniDtaConv_Log.xml 60
security kernel job 43
security kernels 42
server
  deployment 8
server library 18
server statistics screen 27
server workload
  NetBench file server 75
  Samba Server Message Block 75
set number of threads 31
setup
  program 9
  scripts 9
setup operating environment
  instructions for 66
shared disk configuration 114
shared partitions 80
shared tape device 116
simultaneous multi-threading 72
software environment 87
SQL Package 50
SQL Package Library 56
SQL Server import files 9
starting EnterpriseOne services 21
statistics 27
step 1 75
step 2 75
storage cache 114
storage consolidation 117
SUSE LINUX 73
switch 113
switchable device topology 85
switchable IASP 85
switched disk 85
symbolic link 92
synchronous PPRC 109

System Administration Workbench 35
system management resources 111
system performance 25
system value 4
  QCCSID 59
systems management 72

T

table
  data source 93
  F98BLOB 56
  host table entry 98
  OCM 93
  TCP/IP host 98
Table Conversion 55–56
takeover IP 86
tape device
  performance 117
tape devices
  shared 116
TC 56
TCPIP host table 98
technical resources
  PeopleSoft EnterpriseOne 121
third CD 9
threads
  number of 39
  set number of 31
time to save 117
time zone 41
timeout setting 44
Tivoli Performance Viewer 35
Trade3 74
Trade3 application 77
Trade3 workload 73

U

UDB 44
Unicode 55
Unicode ID 60
UniDtaConv.xml 60
UniDtaConv_Log.xml 60
Universal Batch Engine (UBE) job 56
UNIX
  Linux 111
Upgrade Guide 55
upgrade to JD Edwards EnterpriseOne 8.9 55
user profiles
  edit 96

V

V3T 62
vary on deployment server 69
Virtual 3 Tier 62
virtual drive 64
virtual ethernet 77
Virtualization Engine Console 72, 78
Virtualization Grand Slam 72, 75

W

WebSphere 111
WebSphere Administrative Console 37
WebSphere Application Server 5.0 35
WebSphere configuration settings 125
Windows paging file size 8
Windows Server Installation Adviser 66
workload
  AIO 62
  communications 27
  Enterprise Learning Management 73–74
  JAS 62
  native iSeries 111
  Samba 76
  Samba Server Message 75
  server 75
  Trade3 73
  V3T 62

IBM Redbooks

JD Edwards EnterpriseOne 8.9 with IBM @server iSeries and Storage

(0.2”spine)
0.17”<->0.473”
90<->249 pages

JD Edwards EnterpriseOne 8.9 with IBM @server iSeries and Storage

Updated installation tips for JD Edwards EnterpriseOne 8.9 including Unicode

IASP (switched disk) solutions with JD Edwards EnterpriseOne 8.0

WebSphere tuning tips for JD Edwards EnterpriseOne 8.9

JD Edwards EnterpriseOne is a complex application that provides Enterprise Resource Planning (ERP), Supply Chain Management (SCM), and Customer Relationship Management (CRM) capabilities. It can run with Citrix, fat, or Web clients. This IBM Redbook provides helpful hints as you work with the EnterpriseOne product on iSeries.

The reader of this book should have a basic knowledge of IBM @server iSeries and the JD Edwards EnterpriseOne solution. This book is designed to assist technical people among J.D. Edwards OneWorld customers, OneWorld consultants, business partners, and IBM service representatives. It targets professionals who are directly involved with the implementation of a total business solution, consisting of iSeries server hardware, OS/400, DB2 UDB for iSeries database, OneWorld solutions, and supplemental solution products.

You should use this book in conjunction with *J.D. Edwards OneWorld Implementation for AS/400*, SG24-5195. This publication explains detailed concepts and all implementation steps of earlier versions of OneWorld, most of which still apply to OneWorld Xe. You should also use this book with the J.D. Edwards manuals that are provided with the OneWorld software.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6359-00 ISBN 0738490512